Charles River Analytics

Final Report No.: R08098-10
Charles River Analytics Contract No. : C08098
Principal Investigator: Brad Rosenberg
Government Contract No.: N00014-09-C-0050

Enhancing Simulation-based Training Adversary Tactics via Evolution (ESTATE):

Final Report (Option 1)

Brad Rosenberg
James Niehaus
Max Metzger
Scott Neal Reilly

Jordan Pollack
Björn Gunnarsson

Charles River Analytics
625 Mount Auburn Street
Cambridge, MA 02138

Brandeis University
Computer Science Dept
Waltham, MA 024

15 DEC 2010

DISTRUBITION A. Approved for public release; distribution is unlimited

The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the U.S. Government.

Prepared for:
Dr. Harold Hawkins
Office of Naval Research
One Liberty Center
875 N. Randolph Street, Ste 1425
Arlington, VA 22203-1995

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

The public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing the burden, to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.
PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.

1. REPORT DATE *(DD-MM-YYYY)*
15-12-2010

2. REPORT TYPE
Final Report

3. DATES COVERED *(From - To)*
15-12-2009 – 15-12-2010

4. TITLE AND SUBTITLE
Enhancing Simulation-based Training Adversary Tactics via Evolution (ESTATE): Final Report (Option 1)

5a. CONTRACT NUMBER
N00014-09-C-0050

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Brad Rosenberg, James Niehaus, Max Metzger, Scott Neal Reilly

Jordan Pollack, Bjorn Gunnarsson

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Charles River Analytics Inc.
625 Mount Auburn Street
Cambridge, MA 02138

8. PERFORMING ORGANIZATION REPORT NUMBER
R08098-10

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Office of Naval Research
One Liberty Center
875 N. Randolph Street, Ste 1425
Arlington, VA 22203-1995

10. SPONSOR/MONITOR'S ACRONYM(S)
ONR

11. SPONSORING/MONITORING AGENCY REPORT NUMBER

12. DISTRIBUTION AVAILABILITY STATEMENT
DISTRUBITION A. Approved for public release; distribution is unlimited

13. SUPPLEMENTARY NOTES
Report contains color.

14. ABSTRACT
To extend modern simulation-based training environments to incorporate tailored simulation-based training experiences to facilitate accelerated skills development and sustainment, we are developing a system for Enhancing Simulation-based Training Adversary Tactics via Evolution (ESTATE). The system consists of an off-line adaptation engine to extract a model of the trainee based on past performance and generate tailored challenge problems for on-line training. Off-line adaptation is performed using evolutionary algorithms (EAs) to search through the space of challenges to exploit fundamental weaknesses in trainee strategy and tactics. The full-scope prototype ESTATE system is targeting simulation-based training systems within the Deployable Virtual Training Environment (DVTE) to support the squad-level training of U.S. Marines. This report documents the Option 1 period of the effort.

15. SUBJECT TERMS
Behavior modeling, simulation-based training, evolutionary algorithms, coevolution, learning science

16. SECURITY CLASSIFICATION OF:

a. REPORT	b. ABSTRACT	c. THIS PAGE
Unclassified	Unclassified	Unclassified

17. LIMITATION OF ABSTRACT
Unlimited

18. NUMBER OF PAGES
49

19a. NAME OF RESPONSIBLE PERSON
Brad Rosenberg

19b. TELEPHONE NUMBER (*Include area code*)
617-491-3474 x519

Acknowledgment

This work was performed under Government contract number N00014-09-C-0050 with the Office of Naval Research. The authors thank Dr. Harold Hawkins for his support and direction on this project. The authors would also like to thank Ms. Annetta Burger for her continuing responsiveness.

Table of Contents

1. Project Objective 1
2. Project Approach 2
3. Work Completed 3
   3.1 Summary 3
   3.2 Task 2: Develop Mitigation Methods 4
      3.2.1 Disengagement Mitigation, Item Response Theory 4
      3.2.2 Cycling, Overspecialization, Evolutionary Forgetting and Red Queen Effect Mitigation, Coevolutionary Solution Concepts 11
   3.3 Task 3: Enhance Adaptation Techniques 26
   3.4 Task 4: Develop Trainee Model Processing. 26
      3.4.1 Investigation of MoneyBee Data 27
      3.4.2 MoneyBee Player Strategy Visualization 30
      3.4.3 Analysis of Strategy Visualizations 33
   3.5 Task 6: Simulation-based Training System Integration 39
      3.5.1 PROMPTER Overview 39
      3.5.2 ESTATE and PROMPTER Integration 45
   3.6 Task 8: Transition 48
4. References 49

List of Figures

Figure 1: Curve estimation with full range of abilities 9

Figure 2: Learning induced without recalibration 10

Figure 3: Learning induced with recalibration..... 11

Figure 4: An example challenge tree game with k=3, depth=4, and number of goals=4 14

Figure 5: Example game of Nim: <1,2,3,4,5>. Each row represents a heap. Players choose a row and remove one or more bars. The player to remove the last bar wins. 15

Figure 6: MaxSolve implementation on the 3 dimensional discretized COMPARE-ON-ONE game. Student values (in red) increase steadily, and Test values (in blue) maintain a diverse set. 16

Figure 7: Results of MaxSolve on the Challenge Tree game, k=3, d=8, g=10, mutation rate = 1 gene. The top graph shows the mean, minimum, and maximum percentage of winnable nodes that the student population is able to win, graphed by the population generation. As the coevolution progresses, the population improves its ability to win the game. The bottom graph shows the number of tests kept by MaxSolve..... 17

Figure 8: Results of MaxSolve on the Challenge Tree game, k=3, d=8, g=10 (same as Figure 7), *mutation rate = 10 genes*. The top graph shows the mean, minimum, and maximum percentage of winnable nodes that the student population is able to win, graphed by the population generation. As the coevolution progresses, the population improves its ability to win the game. The bottom graph shows the number of tests kept by MaxSolve. 19

Figure 9: Results of MaxSolve on the Nim game. Heaps = <3,3,3,3>. The top graph shows the mean, minimum, and maximum percentage of winnable nodes that the student population is able to win, graphed by the population generation. As the coevolution progresses, the population improves its ability to win the game. The bottom graph shows the number of tests kept by MaxSolve. 20

Figure 10: Results of MaxSolve on the Nim game. Heaps = <3,4,5,4>. The top graph shows the mean, minimum, and maximum percentage of winnable nodes that the student population is able to win, graphed by the population generation. As the coevolution progresses, the population improves its ability to win the game. The bottom graph shows the number of tests kept by MaxSolve. 21

Figure 11: Histogram frequency of output results of 1700 Latin Hypercube samples. 23

Figure 12: Student mutation vs. Output. Test mutation determines the size of the circles. Large student mutation (to the right) and large test mutation (larger circles) produce higher output. ... 24

Figure 13: Dimensionality vs. Output. Student archive size determines the size and color of the circles. Small archives (small, blue circles) are better when dimensionality is small (to the left). Large archives (large, red circles) are better when dimensionality is large (to the right). 25

Figure 14: Log Scaled Heuristic vs. Time to Completion. 28

Figure 15: Estimated problem difficulty per session. 29

Figure 16: Median average game time per session 29

Figure 17: Mean average game time per session 30

Figure 18: Detailed of player strategy graph from problem 1332 31

Figure 19: Entire player strategy graph of problem 1322 32

Figure 20: Strategy Visualization for Session 1 Games 34

Figure 21: Strategy Visualization for Session 2 Games 36

Figure 22: Strategy Visualization for Session 3 Games 37

Figure 23: Strategy Visualization for Session 4 Games 38

Figure 24. (a) Illustration of a casualty with an abdominal wound being laid on their back with their knees bent; (b) PROMPTER icon capturing this body position through a simple, intuitive line drawing. 44

Figure 25. Examples from core set of symbol primitives for commonly occurring objects and actions 44

Figure 26. Examples of compound symbols that combine core symbols of the PROMPTER visual alphabet to express more complex task concepts 44

Figure 27. Examples of rapid prototypes developed to explore and demonstrate promising mechanics for microgame-based learning of procedural knowledge structures, including games that test and develop: (a) knowledge of the meaning of individual symbols; (b) ability to visually recall overarching task mnemonics; and (c) ability to rapidly reconstruct complex task processes 45

Figure 28. Current PROMPTER Architecture 46

Figure 29. ESTATE-PROMPTER simulation framework integration design 47

Figure 30. ESTATE-PROMPTER playable integration design 47

1. Project Objective

To extend modern simulation-based training environments to incorporate realistic and adaptive adversary behavior consistent with today's asymmetric strategies and tactics, we are developing a system for Enhancing Simulation-based Training Adversary Tactics via Evolution (ESTATE). The system consists of: 1) an on-line, executable, reactive adversary behavior model; and 2) an off-line adversary behavior adaptation engine for strategy and tactic discovery. On-line adaptation is performed using an intelligent agent framework to respond and adapt to the trainee's actions during a given simulation-based training exercise. Off-line adaptation is performed using evolutionary algorithms (EAs) to search through the space of adversary behaviors to exploit fundamental weaknesses in trainee strategy and tactics. These adversary behaviors are wargamed against a trainee model extracted from traces of simulation events occurring in past training sessions. The full-scope prototype ESTATE system is targeting simulation-based training systems within the Deployable Virtual Training Environment (DVTE) to support the squad-level training of U.S. Marines.

These objectives have changed from those listed in the original proposal in that we have broadened our scope from adversary behavior to challenges for the trainee that may incorporate the use of adversaries.

DISTRUBITION A. Approved for public release; distribution is unlimited

Contractor Name: Charles River Analytics
Address: 625 Mount Auburn Street, Cambridge, Ma 02138

2. Project Approach

Charles River Analytics and Brandeis University are pursuing an effort to develop and evaluate a full-scope prototype of Enhancing Simulation-based Training Adversary Tactics via Evolution (ESTATE), a tool to provide tailored training in line with irregular warfare for synthetic training environments. The proposed project consists of the following tasks: Task 1: Identify Training Goals, Task 2: Develop Mitigation Methods, Task 3: Enhance Adaptation Techniques, Task 4: Develop Trainee Model Processing, Task 5: Develop Tools for Intelligent Agents, Task 6: Simulation-based Training System Integration, Task 7: Evaluation and Demonstration, Task 8: Transition, and Task 9: Documentation and Reporting.

This approach has not changed from that listed in the original proposal, aside from relaxing terminology used to broaden scope to challenges, vice adversaries.

3. Work Completed

3.1 Summary

The primary focus of this reporting period was on Task 2: Develop Mitigation Methods, Task 3: Enhance Adaptation Techniques, Task 4: Develop Trainee Model Processing, Task 6: Simulation-based Training System Integration, and Task 8: Transition.

Under Task 2, we have investigated, implemented, and tested a monotonic solution concept, MaxSolve (De Jong, 2005), and applied Item Response Theory (IRT) (Baker, 2001) to mitigate key coevolutionary pathologies.

Under Task 3, we have applied IRT to select challenges based on estimation of the trainee's skill, and we have applied strategy-based coevolution to select challenges that fall within the trainee's Zone of Proximal Development (ZPD) based on estimation of the trainee's strategy.

Under Task 4, we have analyzed the MoneyBee data set to discover how trainees may develop skill and learn strategies. We have applied IRT to address key trainee model processing issues of bootstrapping, self-sufficiency, and dynamics.

Under Task 6, we have initiated integration of the ESTATE prototype into an existing microgame-based training platform.

Under Task 8, we have pursued opportunities for transition with U.S. Marine Corps Training & Education Command and PM Training Systems.

Our accomplishments during the current reporting period have made use of two perspectives on trainee ability. Item Response Theory (IRT) (Baker, 2001) treats ability as a *scalar value* relating to a particular skill (e.g. a 5 out of 10 on combat-hunter skills). Our investigations into IRT have provided critical information about the data collection needs and dynamics of challenge-based tailored training. However, we believe a single scalar value to be insufficient for representing trainee ability in the complex domains such as irregular warfare, cultural training, and combat-hunter skills. These domains often have many possible courses of action that lead to desirable outcomes, and simply understanding that a trainee is moderately skilled does not provide concrete avenues for assessment, training, and improvement.

To provide tailored training in complex domains, a training system must reason about where a trainee's weakness lie and the circumstances under which the trainee performs poorly. Our current coevolutionary approach represents trainee ability as a *strategy*, a mapping of world states to actions. A strategy prescribes what a trainee will do if presented with a particular situation or particular type of situation. Strategies can represent behavior that is complex and

nuanced, relying upon a number of situational dimensions to generate behavior, and this representation can be used to identify specific dimensions of trainee weaknesses to be addressed.

Below, we describe our findings using the IRT model, our findings in the current coevolutionary strategy model, and how we combine the two sets of findings to present a broader view of challenge-based tailored training.

3.2 Task 2: Develop Mitigation Methods

The purpose of Task 2 is to mitigate the effects of coevolutionary pathologies on trainee progress with ESTATE. Applying coevolution techniques often fails to generate the desired goal of a continuous learning process leading to ever-improving individuals. Recent research has begun to identify and define the pathologies hindering success. With the assistance of our university partner, we have identified key pathologies to address:

- *Disengagement*: Occurs when one population (challenges or students) is consistently superior to the other. Loss of competitive gradient causes improvement to cease.
- *Cycling or Intransitivity*: Oscillation back and forth between strategies causes overall improvement to cease.
- *Overspecialization or Focusing*: Concentration in one area at expense of other areas causes brittle strategies that do not perform well in all circumstances.
- *Evolutionary Forgetting*: Loss of useful trait from one generation to the next, causes cycling or strategy degradation over time.
- *Red Queen Effect*: Changes which improve quality of a solution do not increase its selection probability due to changes to other coevolving solutions. May cause strategies to wander randomly and often degrade.

Based on the many common pitfalls facing coevolving systems, we strongly believe that any approach that fails to address these competitive pathologies will be unknowingly subject to failure. Our approach identifies methods to mitigate these pathologies and thus improve training gains.

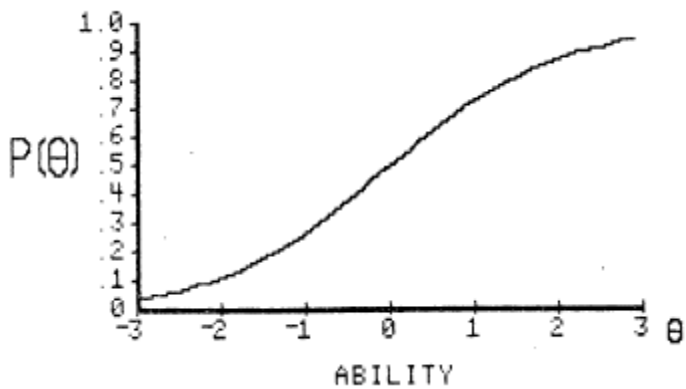
3.2.1 Disengagement Mitigation, Item Response Theory

Disengagement occurs when one population in coevolution is consistently superior to the other. In ESTATE, this occurrence would indicate that either 1) the trainees are far in advance of the challenges and the challenge generator cannot find a challenge that is difficult enough, or 2) the challenges are too difficult for the trainees and they cannot make incremental steps toward improving their ability. Disengagement can be mitigated by accurately estimating the trainee's

ability and generating a challenge to meet or barely exceed that ability. One method to estimate trainee ability is with IRT, using the item characteristic curve.

3.2.1.1 Item Characteristic Curve

In IRT, *ability* is used to represent and measure latent traits in individuals performing a function. We represent this term by θ. While θ can range from positive infinity to negative infinity, it is typically given a -3 to 3 range. For each item (or challenge), an individual has a probability of getting the item correct or incorrect. This probability is represented by P(θ). Since P(θ) is a function of θ, we can construct an *item characteristic curve* (ICC) that represents the probability of getting an item correct as a function of an individual's ability level. These ICCs are normally S-curves. The shape of these S-curves can be defined by several mathematical models. The *difficulty* of an item is a location index that describes where the item functions along the ability scale. For our purposes, this can be where is P(θ) = 50%. The *discrimination* of an item describes how well the item can differentiate between examinees having abilities below the item location and those having abilities above the item location (essentially the steepness of the ICC in the middle, or the slope of the line where P(θ) = 50%). The *guessing* of an item describes how likely it is that an examinee will guess the answer correctly.



The equation for the three parameter ICC (Baker, 2001) is:

$$P(\theta) = c + (1-c)\frac{1}{1+e^{-a(\theta-b)}}$$

Where: *b* is the difficulty parameter
a is the discrimination parameter
c is the guessing parameter and

θ is the ability level

Note that in simulation, a response may be generated from this equation by setting a response value r such that: $r = P(\theta) < U(0,1)$; where $U(0,1)$ is a random number from the uniform distribution between 0 and 1, inclusive.

The single parameter model, or Rasch model, is defined as the above ICC with a=1.0 and c=0.

3.2.1.2 Estimating an Examinees Ability

Given a set of ICCs and a history of results for an examinee, it is possible to estimate the examinees ability. The estimation equation for maximum likelihood is:

$$\hat{\theta}_{s+1} = \hat{\theta}_s + \frac{\sum_{i=1}^{N} a_i \left[u_i - P_i(\hat{\theta}_s) \right]}{\sum_{i=1}^{N} a_i^2 P_i(\hat{\theta}_s) Q_i(\hat{\theta}_s)}$$

where: $\hat{\theta}_s$ is the estimated ability of the examinee at iteration s

a_i is the discrimination parameter of item i

u_i is the response made by the examinee to item i: 1 for correct, 0 for incorrect

$P_i(\hat{\theta}_s)$ is the probability of a correct response to item i under the given item characteristic curve.

$Q_i(\hat{\theta}_s) = 1 - P_i(\hat{\theta}_s)$ is the probability of an incorrect response

Thus, a running estimate of an examinee's ability can be computed in simulation by computing the adjustment after each item result. Note that if the examinee answers either all or none of the items correctly then the estimation is either infinity or division by zero respectively.

3.2.1.3 Applying Item Response Theory to ESTATE

Using Item Response Theory, we can think of the ESTATE conceptual formulation in another way. A trainee has an ability level at any given time, represented by θ. Since we can never know the true ability of the trainee, we can only estimate it. This estimation is assigned $\hat{\theta}_s$. Via simulation, we can bring the trainee ability against a challenge c and produce a result r. We build up a repository of these interactions as a history of tuples $< c_i, \hat{\theta}_i, r_i >$. During diagnosis, we assess the current estimated ability level of the trainee based on the history of traces and

determine $\hat{\theta}_s$. During adaptation, we attempt to find the optimal challenge c^* that will promote learning to serve the next round. c^* can be derived from finding the challenge such that the probability of getting that challenge correct given the currently estimated ability level of the trainee is greater than or equal to the probability of getting that challenge correct at the optimal ability level minus some delta. Formally, $P_{c*}(\hat{\theta}_i) + \Delta P >= P_{c*}(\theta^*)$. We can assume that $P_{c*}(\theta^*) = 0.5$, since at the target ability level, with the optimal challenge, the trainee has a 50% chance of responding to the challenge correctly. Furthermore, we can start with ΔP at 5% or 10% as an assumption of the zone of proximal development (ZPD). We can then adapt ΔP based on the current trend in answers being correct or incorrect in recent history. Based on this, $60\% >= P_{c*}(\hat{\theta}_i) >= 40\%$ with a $\Delta P = 10\%$.

3.2.1.4 Key Issues

We have identified three key issues that arise from using IRT to estimate trainee ability during challenge-based tailored training: bootstrapping, self-sufficiency, and dynamics.

1) **Bootstrapping**: Given the model above, $\hat{\theta}_s$—the estimate of the trainee's ability—must be within a small error to derive a challenge problem that will fall in the ZPD and stimulate learning. ESTATE's estimated ability of the trainee must be close enough to the trainee's actual ability to be able to formulate a problem that is appropriately challenging. How many challenges must the trainee attempt before $\hat{\theta}_s$ falls within this error? This number must be small enough to reasonably require the trainees to attempt this many challenges before receiving learning gains from the system.

2) **Self-Sufficiency**: The input to the system should be as little as possible. Defining a curriculum of challenges, determining their difficulty, and ranking the abilities of training are extremely difficult and time consuming tasks for a training instructor and system developer. ESTATE should structure interactions to gather as much of this information as possible. Ideally, ESTATE should be given only a set of features used to create challenges and a scoring mechanism. The system should be able to assess trainees' ability and promote learning.

3) **Dynamics**: Traditional item response theory does not account for the possibility of learning as a result of attempting items. However, we expect the challenges in ESTATE to promote learning in the trainees. ESTATE must predict or assess learning gains to prevent its estimates of a trainee's ability from becoming inaccurate over time. ESTATE must balance choosing learning challenges with choosing assessment challenges.

3.2.1.5 Estimating both the Challenge Curve and Trainee Ability

Since ESTATE may be generating the challenges that trainees attempt, we cannot assume that we will have a well-defined challenge curve for each challenge. ESTATE must estimate both the trainee's ability and the challenge curve simultaneously. Since the ICC depends on the estimate of ability and the estimate of ability depends on the performance from an ICC, ESTATE must make an assumption about either the abilities of the trainees or the shape of the challenge curve. In the case where the challenge curve cannot be assumed, assumptions about the trainees' abilities may be made. Because the trainees' abilities are due to a large number of possible factors, the central limit theorem indicates that the abilities may be assumed to be normally distributed – such an assumption is often used initially for data concerning human performance. Thus, the shape of the challenge curve can be estimated from the set of scores.

First the estimated points on the ability/score graph will be computed, then a spline curve will be used to interpolate the function representing these points. We make the additional assumption that the challenge curve is monotonically increasing: higher displayed ability will result in an equal or higher score. The scores are ordered by increasing value, and the abilities are calculated as if constructing a normal probability plot:

$$\hat{\theta} = f(x) = G(U(x))$$

where *U(x)* are the uniform order statistic medians
G(x) is the percent point function (inverse of the cumulative distribution function) of the normal distribution

A cubic spline may be interpolated from these points to create an estimate of the challenge curve. The details of this interpolation are beyond the scope of this document.

Figure 1 presents the results of one such estimation. 20 trainees with abilities sampled from a normal distribution, $N(\mu = 0, \sigma^2 = 1.0)$, each attempt a challenge, displaying ability with a small variance from their actual ability ($\sigma^2 = 0.1$). As is evident from the figure, the challenge curve is estimated with a high degree of accuracy (average error = 0.018%).

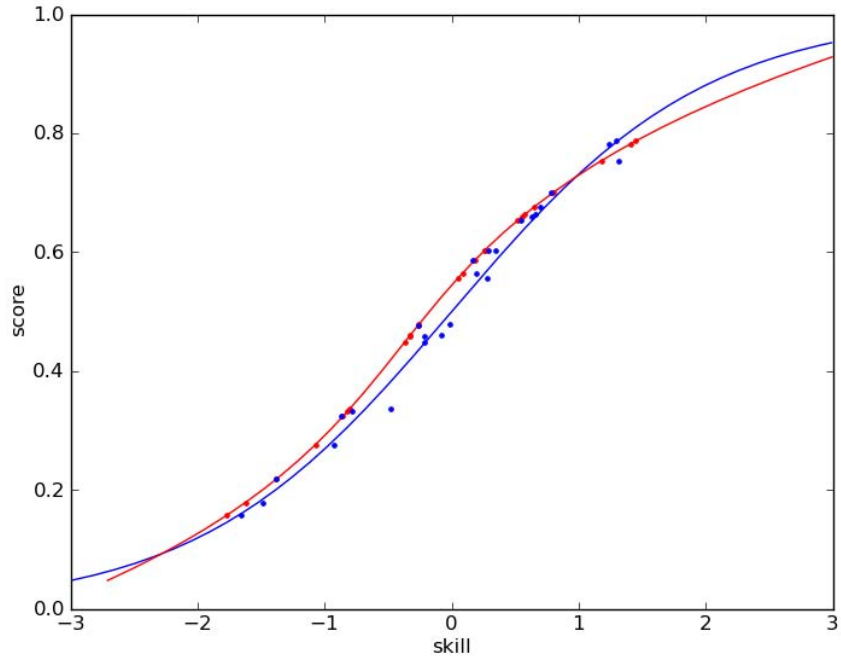


Figure 1: Curve estimation with full range of abilities

The blue line is the actual challenge curve. The blue points are simulated scores with displayed ability normally distributed about actual ability (std dev = 0.1). The red line is estimated challenge curve. The red points are the estimated skill for each score. 20 trainees, 1 attempt each. Error is 0.018%

3.2.1.6 Continuous Estimation and Learning

The challenge curve estimation above does not yet consider learning over time. How does learning affect the accuracy of the estimate? Can ESTATE promote continuous learning using the above approach? We measure the effectiveness of this approach in simulation.

Given a set of low ability trainees and a set of challenges with a full range of difficulty, can ESTATE reliably target trainees' ZPD and promote learning over time? Our simulation is initialized with a group of trainees with abilities averaging -2.5 on a [-3,3] scale (std dev is 0.15) and a set of 100 challenges (using the Rasch model) with difficulties spaced equally along the same range. A trainee's skill will improve by a small increment, 0.05, if his expected score is between 60% and 70%, the ZPD for this simulation. Given the parameters above, there is always at least one 'correct' challenge to present to a trainee. The simulation estimates the challenge curve based upon the history of scores. The next challenge for a trainee is chosen by finding the

challenge with the expected score, based on the estimated curve, that falls within the range above.

Figure 2 presents the results of this simulation. After the initial estimation of the curve, the choice of challenge briefly matches the theoretical best choice. At about the 7th round of challenges, the estimate begins to depart sharply from the best choice. At about the 14th round of challenges, the estimate is no longer able to choose a challenge in the ZPD, and the learning of the trainees is halted.

These results occur because the trainee's abilities climb out of the range of the estimated challenge curve. The challenge curve is attempting to estimate a score for an ability for which it has not yet seen. In order to provide an accurate estimate, the curve needs to be calibrated not only once, but after learning occurs. Figure 3 presents the same simulation if recalibration is introduced after every 7th round. Here, the estimated result keep pace with the learning of the trainees, and the choices based on the estimate follow closely with the theoretically best choices.

As Figure 3 indicates, *ESTATE can use its estimation of trainee's abilities to promote continuous situational learning.* If the abilities of the trainees are normally distributed, *ESTATE can automatically discover the challenge appropriate for a particular trainee at a particular time, reducing the effects of disengagement.*

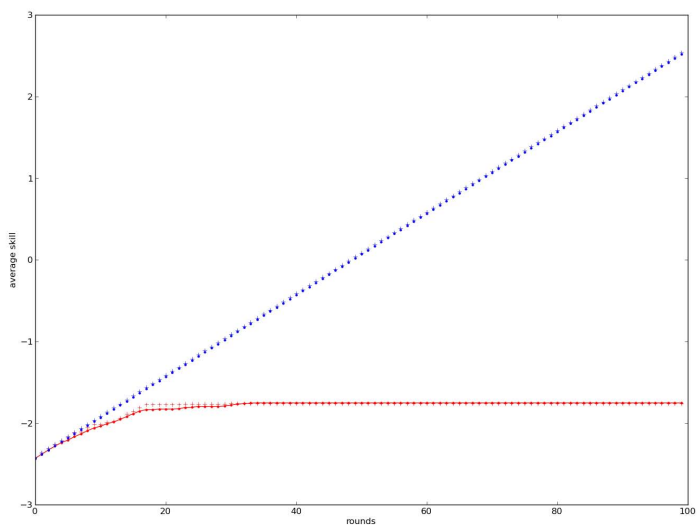


Figure 2: Learning induced without recalibration

Filled points are the mean ability, '+' points are the median ability. Blue points are theoretical best choice. Red points are chosen using estimated values.

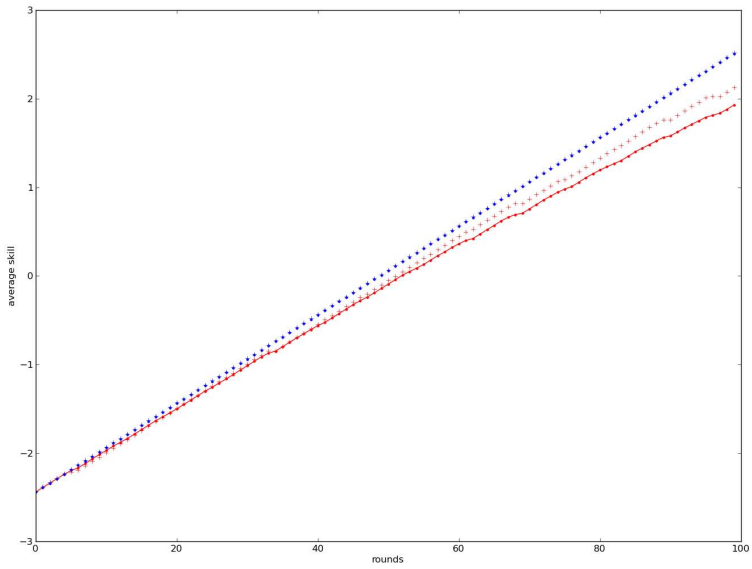


Figure 3: Learning induced with recalibration

Filled points are the mean ability, '+' points are the median ability. Blue points are theoretical best choice. Red points are chosen using estimated values.

3.2.2 Cycling, Overspecialization, Evolutionary Forgetting and Red Queen Effect Mitigation, Coevolutionary Solution Concepts

During the current reporting period, we examined the use of the following techniques for mitigating coevolutionary pathologies when representing trainee ability as a strategy.

- Capturing Informativeness – Mitigating disengagement by identifying how solutions can inform (e.g., test) other competing solutions and maintaining them in the population based on this criteria.
- Separation of Teacher and Learner Populations – Capturing informativeness by explicitly separating the population of strategies into two populations, one that informs the evolutionary algorithm on how the other one is doing.
- Memory Mechanisms – Improving upon standard elitism using "Hall of Fame" techniques to provide a growing external benchmark to compare newer potential solutions against older potential solutions, mitigating the cycling, overspecialization, evolutionary forgetting, and red queen effect pathologies.

To capture informativeness and separate teacher and learner populations, we employ student-test coevolution. Our coevolution simulations consist of two populations. One population, the tests, represents the challenges that may be presented to the trainees. The other population, the students, represents strategies that the trainees may use to attempt the challenges. Because the purpose of student-test coevolution is to produce better students and the quality of the tests are only important in their ability to improve student performance, different selection strategies are used for students and tests. Students are chosen according to their ability to pass all of the current tests. The highest performing group of students is usually chosen for the next generation. Tests, however, are chosen according to their ability to inform on the students progress. A test that defeats all students is not as useful to the algorithm as one that defeats only half of the students. The second test provides information about which students are better, and thus should be generally preferred to the first test.

3.2.2.1 Selecting a Solution Concept for Student-Test coevolution

Ficici (2004) identifies *solution concepts* as a method to analyze the relationship between the selection of individuals in coevolution and the meeting of the overall goals of the coevolutionary process. It indicates which individuals to keep for future populations; thus, a solution concept is a type of memory mechanism. A well-functioning solution concept will drive the population towards the goals (e.g. being a better game player), while a poorly functioning solution concept will cause the population to flounder due to one or more coevolutionary pathologies.

A *monotonic* solution concept (Ficici, 2004) is one that causes the best individual in the population to drift no further from the goal, with some chance of evolving towards the goal. Thus the population monotonically increases its fitness according to the goal. A monotonic solution concept prevents the pathologies of cycling and intransitivity, evolutionary forgetting, and the red queen effect from occurring during coevolution. To avoid these pathologies while still allowing execution within a reasonable time, we chose and implemented a method to approximate a monotonic solution concept for student-test coevolution.

Several general purpose monotonic solution concepts have been identified in the literature. Rosin (1997) identifies a solution concept that selects students that simultaneously maximize outcomes over all tests. However, we do not expect our challenges to allow a single, correct solution at each level. Ficici (2004) identifies a solution concept according to the Nash equilibrium, where no individual can individually change his strategy without decreasing his payoff, i.e. there is no individual incentive to change. The IPCA algorithm (De Jong, 2004) identifies a solution concept based on the Pareto-optimal equivalence set, a set that provides

maximal trade-off between objectives. MaxSolve (De Jong, 2005) identifies a solution concept based on the maximum expected outcome of a student over the current population of tests. Finally, DECA (De Jong & Bucci, 2006) identifies a solution concept based on estimating the true problem dimensionality, the number of different objectives which must be maximized simultaneously.

Our criteria for selecting a solution concept was that 1) the solution concept performed well in practice and 2) the solution concept did not further constrain on the problem. ESTATE's effectiveness as an adaptive training environment will be increased the better a solution concept is able to perform. New challenges will be created more quickly, and they will be closer to the optimal Zone of Proximal Development of the trainee. Secondly, we did not wish to overly constrain the problem that our coevolutionary technique can address. Such constraints may prevent training of critical skills, and we wish for ESTATE to be applicable to as many skill sets as possible.

Performance comparisons between these algorithms (De Jong, 2005; De Jong et al., 2006), communications with authors (Bucci, 2010), and consultation with our academic partner, an expert in this area, led us to choose the MaxSolve solution concept as the best candidate for implementation and testing. MaxSolve has exhibited high performance on a number of different challenges, and it does not place any additional constraints on the problem.

3.2.2.2 Implementing MaxSolve and Challenge games

To evaluate the performance of the MaxSolve solution concept for use in ESTATE, we implemented student-test coevolution, the solution concept, and several test problems. We leveraged our in-house evolutionary algorithms toolkit, EAToolkit, and expanded the toolkit to support competitive coevolution, in which individuals are scored according to one-on-one competitions, and student-test coevolution, in which a separate student and test populations are maintained. Tests are challenges that may be presented to a trainee, and students are strategies for overcoming challenges. Ideally, coevolution will result in challenges of increasing difficulty and strategies of increasing effectiveness.

We implemented the MaxSolve solution concept as described in (De Jong, 2005). To ensure a correct implementation and provide an assessment of performance, we implemented three separate games for coevolution testing.

The first game implemented was the discretized COMPARE-ON-ONE numbers game from the MaxSolve paper (De Jong, 2005). The numbers game is a simple game where the individuals

attempt to increase the values of their vectors of real numbers. Both students and tests are individuals with a vector of numbers. When a student attempts a test, the individual with the higher value in the slot with the test's highest value wins. This game is advantageous in that it provides a difficult test case for coevolution (because the simple mechanics are a black box to the coevolutionary algorithm) while remaining open to rapid analysis.

The second game implemented was the challenge tree game, intended to mimic the structure of an actual strategy space that may be input to ESTATE. A challenge tree is a complete *k*-ary tree of depth *d*. Each non-leaf node in the tree has *k* children and the path from the root node to a leaf node is of length *d*. A number, *g*, of the leaf nodes are identified as goal states. Figure 4 is an example challenge tree with *k*=3, *d*=4, and *g*=4. A challenge tree can be played by beginning at the root and choosing a child node to move to until a leaf node is reached. If the leaf node is a goal state, then the game was won, else the game was lost. In student-test coevolution, tests are sub-trees (beginning nodes) of a larger challenge tree, and students are strategies consisting of <node, child-node> pairs specifying which child node to choose at each node.

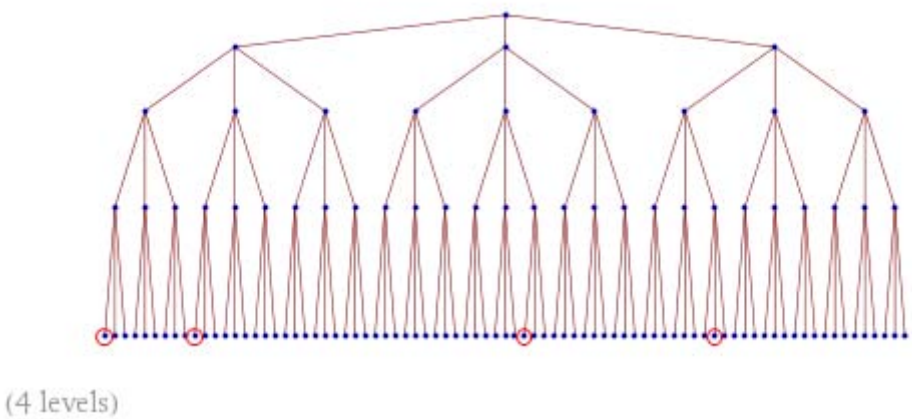


Figure 4: An example challenge tree game with k=3, depth=4, and number of goals=4

The third game implemented was the game of Nim. This game was intended to test the coevolution on an actual game that humans find challenging to play despite the existence of a relatively straightforward perfect strategy (Bouton, 1901). Nim is played with *n* heaps of stones of varying sizes. Each player takes turns selecting a heap, then picking up one or more stones from that heap. The player to pick up the last stone wins. Figure 5 is an example game of Nim with heap sizes 1, 2, 3, 4, and 5. In student-test coevolution, tests are initial game states, and students are strategies consisting of <state, action> pairs specifying which heap to select and how many stones to remove from this heap. Because Nim is a 2-player game, we pit the students against an automated player with the perfect strategy (most games can still be won, as the student

makes the first move). This formulation is particularly restrictive because any deviation from the perfect strategy within the sub-game will result in a loss.

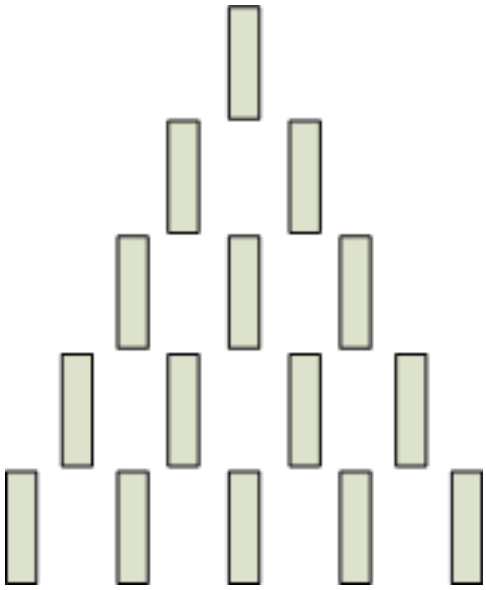


Figure 5: Example game of Nim: <1,2,3,4,5>. Each row represents a heap. Players choose a row and remove one or more bars. The player to remove the last bar wins.

3.2.2.3 Testing performance of MaxSolve coevolution

The first test of our MaxSolve implementation was to reproduce the results of the original paper using the discretized COMPARE-ON-ONE game (De Jong, 2005). Figure 6 shows the results of this test. These results match those of the paper; MaxSolve is able to sustain continuous student improvement on three dimensions by maintaining a diverse set of tests. This confirms the results of the paper and supports our claim of correct implementation.

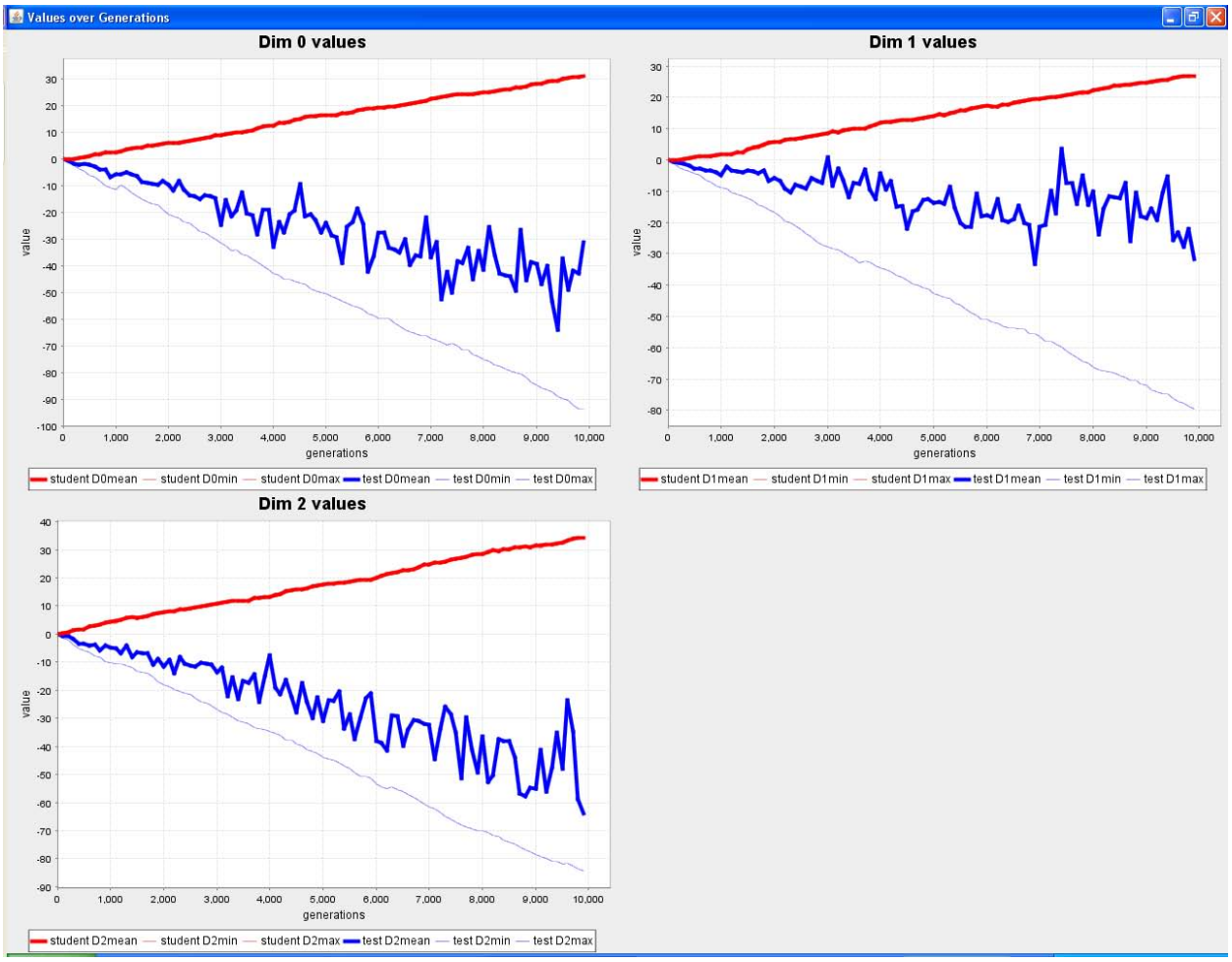


Figure 6: MaxSolve implementation on the 3 dimensional discretized COMPARE-ON-ONE game. Student values (in red) increase steadily, and Test values (in blue) maintain a diverse set.

The second test was to use MaxSolve in the challenge tree game. Figure 7 shows the results of challenge tree coevolution with k=3, d=8, g=10, and the mutation rate for students set to 1 gene (one node's strategy is changed for each child). The best students are able to find a goal state for 57% of the winnable nodes by generation 1000.

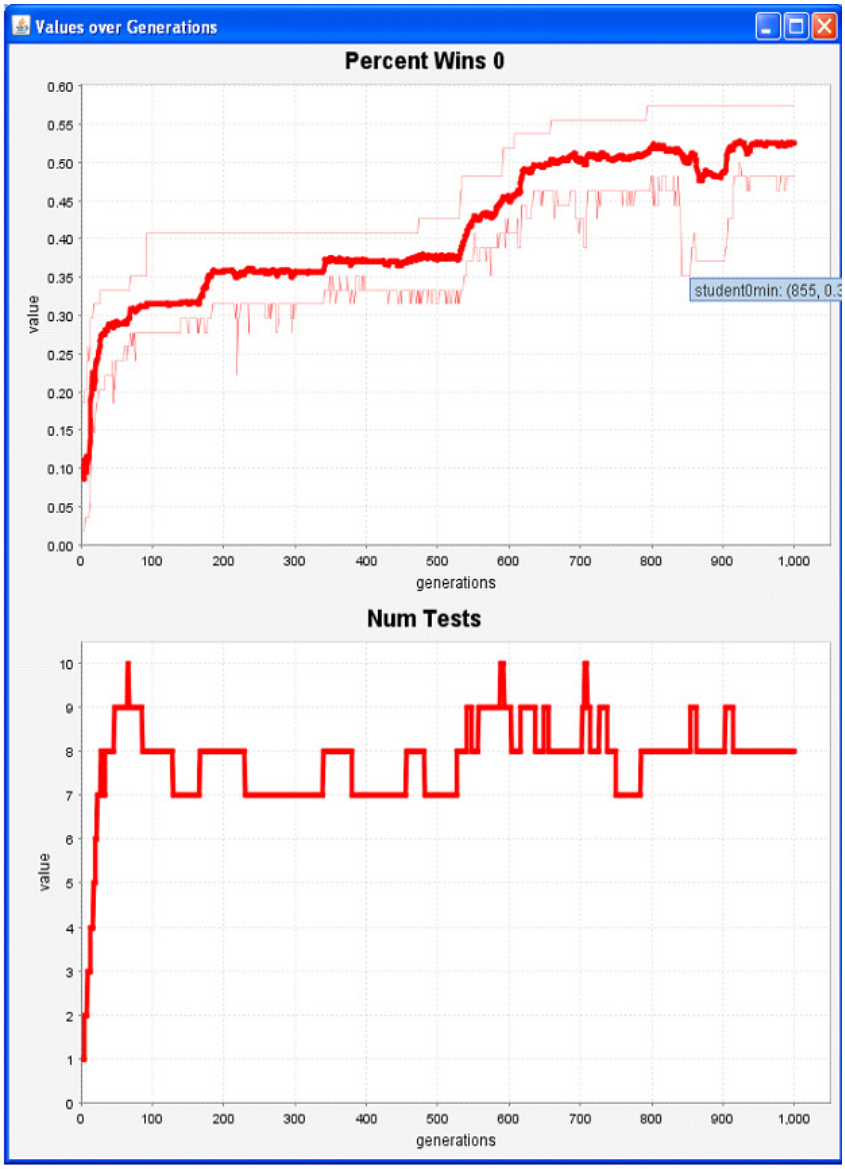


Figure 7: Results of MaxSolve on the Challenge Tree game, k=3, d=8, g=10, mutation rate = 1 gene. The top graph shows the mean, minimum, and maximum percentage of winnable nodes that the student population is able to win, graphed by the population generation. As the coevolution progresses, the population improves its ability to win the game. The bottom graph shows the number of tests kept by MaxSolve.

One of the issues in applying coevolutionary solutions to problems such as this is the tuning of algorithm parameters to improve performance. Parameters in this formulation of student-test coevolution are student mutation rate, test mutation rate, student crossover percentage (the percent of new individuals are created through crossover), test crossover percentage, student archive size, student population size, test population size, and initial population sizes. As an

example of how these parameters may contribute to the effectiveness of the coevolution, Figure 8 shows the results of increasing the student mutation rate to 10 genes. Here, the best students are able to find a goal state for 99% of the winnable nodes by generation 1000, a substantial improvement over the previous run. To provide some insight into optimal parameter settings, we performed a sensitivity analysis MaxSolve coevolution of the COMPARE-ON-ONE game; the results are summarized in Section 3.2.2.4. These results show that MaxSolve can be effective in strategy domains such as those ESTATE may encounter, given proper parameter settings.

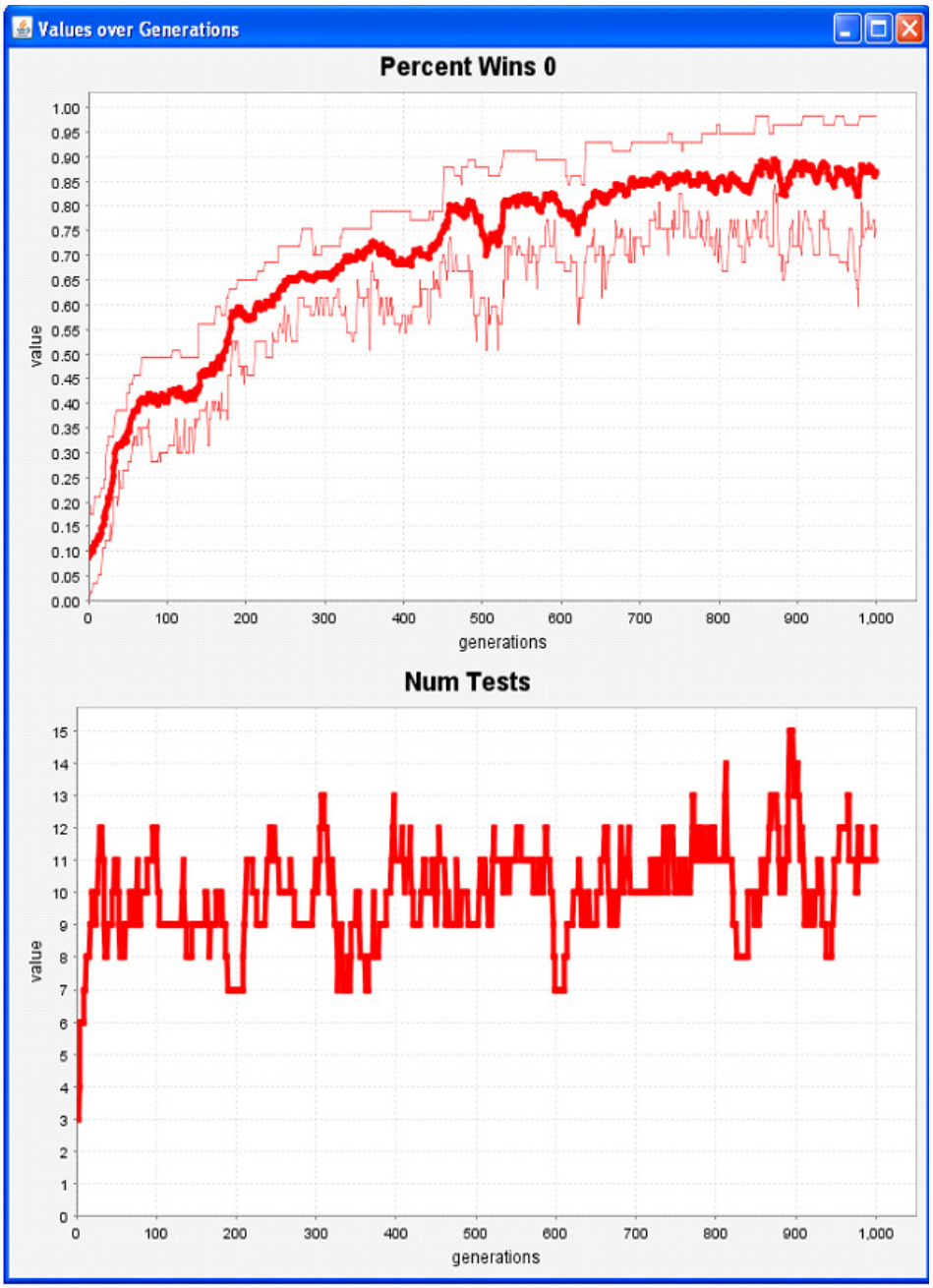


Figure 8: Results of MaxSolve on the Challenge Tree game, k=3, d=8, g=10 (same as Figure 7), *mutation rate = 10 genes*. The top graph shows the mean, minimum, and maximum percentage of winnable nodes that the student population is able to win, graphed by the population generation. As the coevolution progresses, the population improves its ability to win the game. The bottom graph shows the number of tests kept by MaxSolve.

Neither the challenge tree game nor the COMPARE-ON-ONE game are difficult for humans to learn or solve, the next test was to apply MaxSolve to the Nim game to test performance on a

larger problem that humans do not easily solve. In this regard, Nim is representative of the aims of ESTATE, to aid trainees in developing real world skills on difficult problems and tasks. Figure 9 shows the results of running MaxSolve coevolution on the Nim game with heap sizes = <3,3,3,3>. Here, coevolution is again successful in finding a winning strategy after about 600 generations; the best student is able to win against the perfect player at any winnable sub-game – it has evolved the perfect strategy.

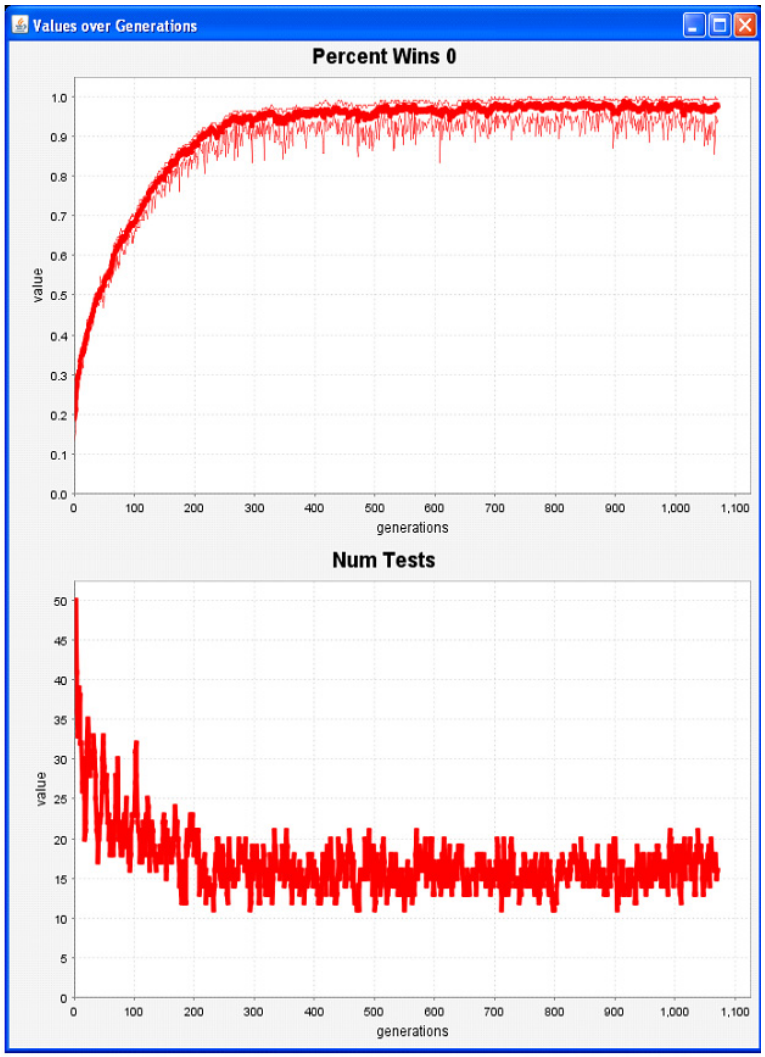


Figure 9: Results of MaxSolve on the Nim game. Heaps = <3,3,3,3>. The top graph shows the mean, minimum, and maximum percentage of winnable nodes that the student population is able to win, graphed by the population generation. As the coevolution progresses, the population improves its ability to win the game. The bottom graph shows the number of tests kept by MaxSolve.

Nim is made more difficult for coevolution by increasing the size of the piles. Figure 10 shows the results of running MaxSolve coevolution on Nim with heap sizes = <3,4,5,4>, more

than doubling the size of the state space. Here, the coevolution takes much longer to converge on a successful strategy, but the best player is still able to win 95% of the winnable games against the perfect player after 10,000 generations. These results show good performance of MaxSolve coevolution on a game that novice humans have difficulty winning consistently. This is a strong indication that our MaxSolve student-test coevolution will be able to make progress in domains that require non-trivial strategic formulations, such as those training domains that ESTATE targets.

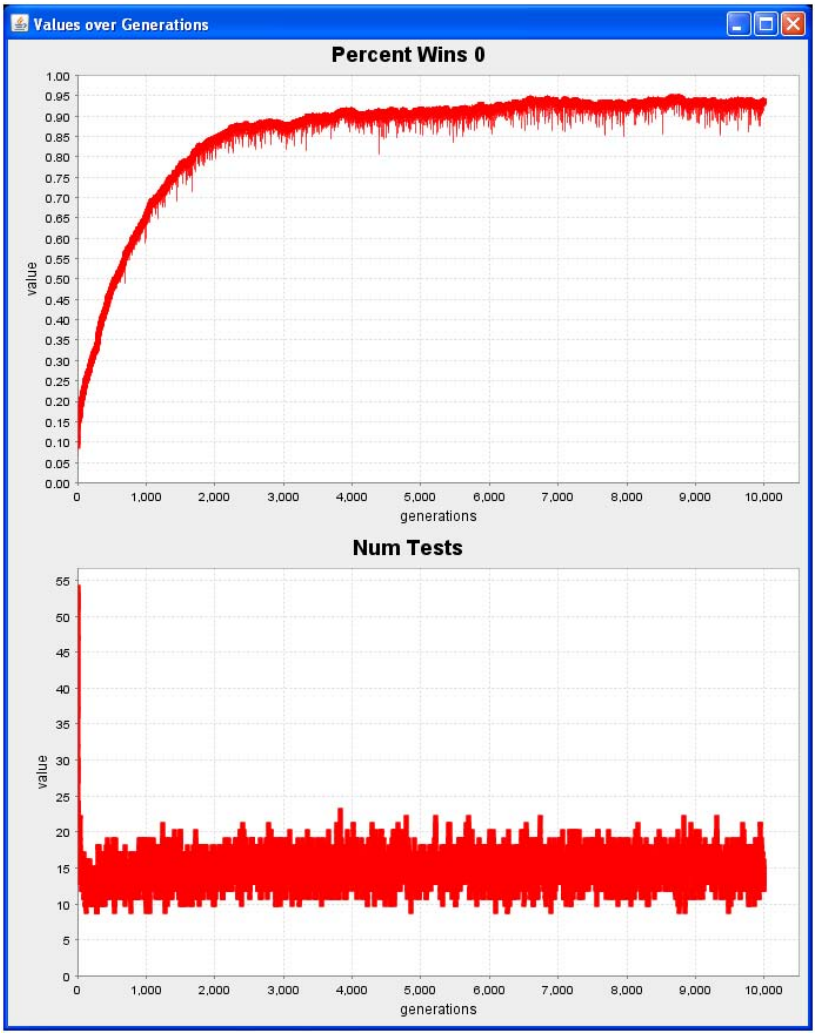


Figure 10: Results of MaxSolve on the Nim game. Heaps = <3,4,5,4>. The top graph shows the mean, minimum, and maximum percentage of winnable nodes that the student population is able to win, graphed by the population generation. As the coevolution progresses, the population improves its ability to win the game. The bottom graph shows the number of tests kept by MaxSolve.

Together, these results show strong support that MaxSolve can produce successful coevolution on a range of different, yet representative, problems, *mitigating the cycling, overspecialization, evolutionary forgetting, and red queen effect pathologies.*

3.2.2.4 Sensitivity Analysis of MaxSolve Coevolution

One of the issues noted above in using coevolution is the tuning of algorithm parameters to improve performance. As the challenge tree example in Section 3.2.2.3 exhibits, choosing optimal parameters can make the difference between success and failure. Parameters in our MaxSolve student-test coevolution are student mutation rate, test mutation rate, student crossover percentage, test crossover percentage, student archive size, student population size, test population size, and initial population sizes. Also to be considered is the difficulty of the problem under consideration. Here, we perform a sensitivity analysis of problem dimensionality (number of simultaneous objectives, roughly a measure of difficulty), MaxSolve archive size, student mutation rate, test mutation rate, and crossover percentage for the discretized COMPARE-ON-ONE numbers game as defined in Section 3.2.2.2. The sensitivity analysis indicates how these parameters interact to produce a change in the result.

1700 samples of the parameter space were created using a Latin Hypercube design. Student archive size ranged from 10 to 160, dimensions ranged from 2 to 10, student and test mutation rates ranged from 0.05 to 0.75, student crossover percentage ranged from 0.5 to 0.75. The output variable was the mean of the allele vector of the best student in the population (each individual is a vector of numbers), approximately the average “goodness” of the top student. Figure 11 shows the frequency of the output as a result of these samples; most of the results were within the 0-7 range, with a few outliers. The COMPARE-ON-ONE game chooses the higher of two individuals, thus higher output is better.

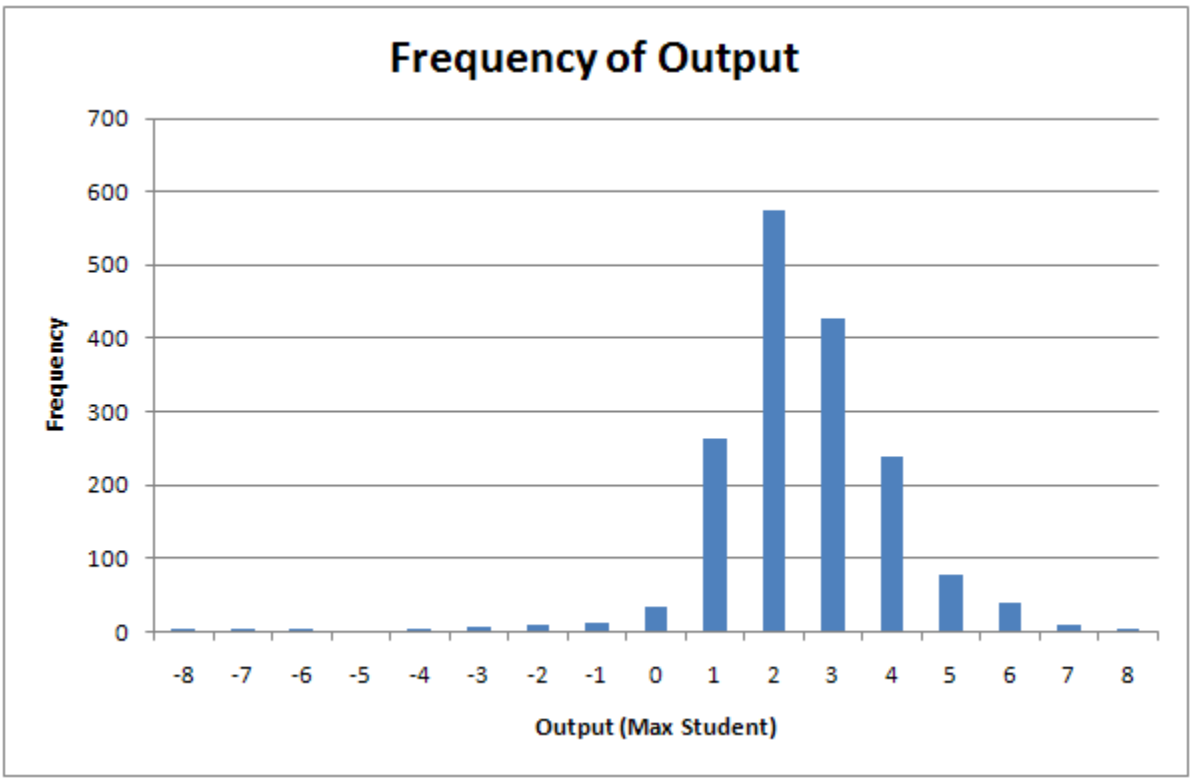


Figure 11: Histogram frequency of output results of 1700 Latin Hypercube samples.

Two strong relationships emerged from the analysis. The first is that when both student and test mutation were high, the result was high. The second is that the optimal student archive size depends on the dimensionality of the problem. For problems of low dimensionality, a small archive is best, larger archives produce worse results, for problems of high dimensionality, a large archive is best, larger archives produce better results. Figure 12 and Figure 13 show plots of the samples indicating these results.

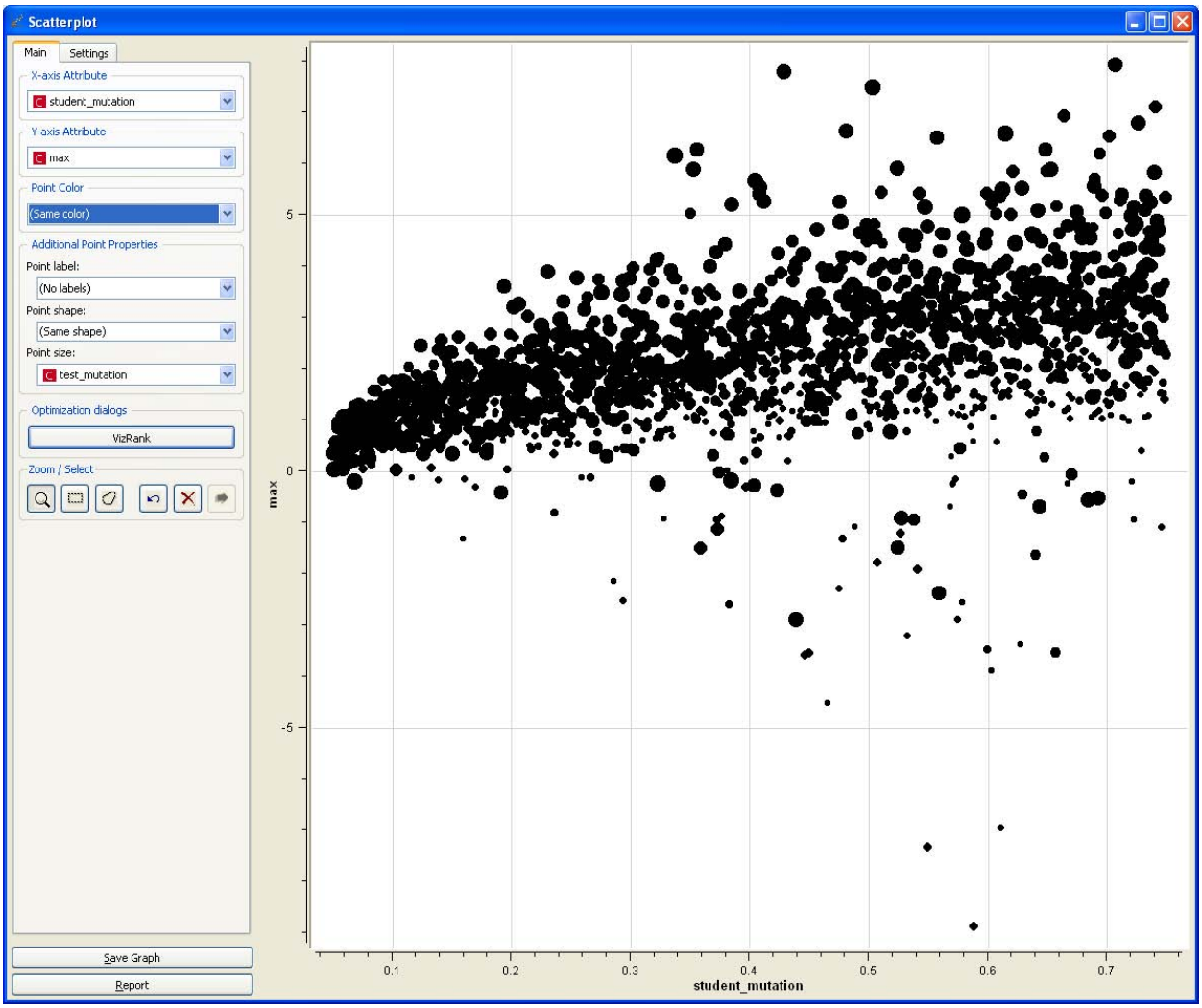


Figure 12: Student mutation vs. Output. Test mutation determines the size of the circles. Large student mutation (to the right) and large test mutation (larger circles) produce higher output.

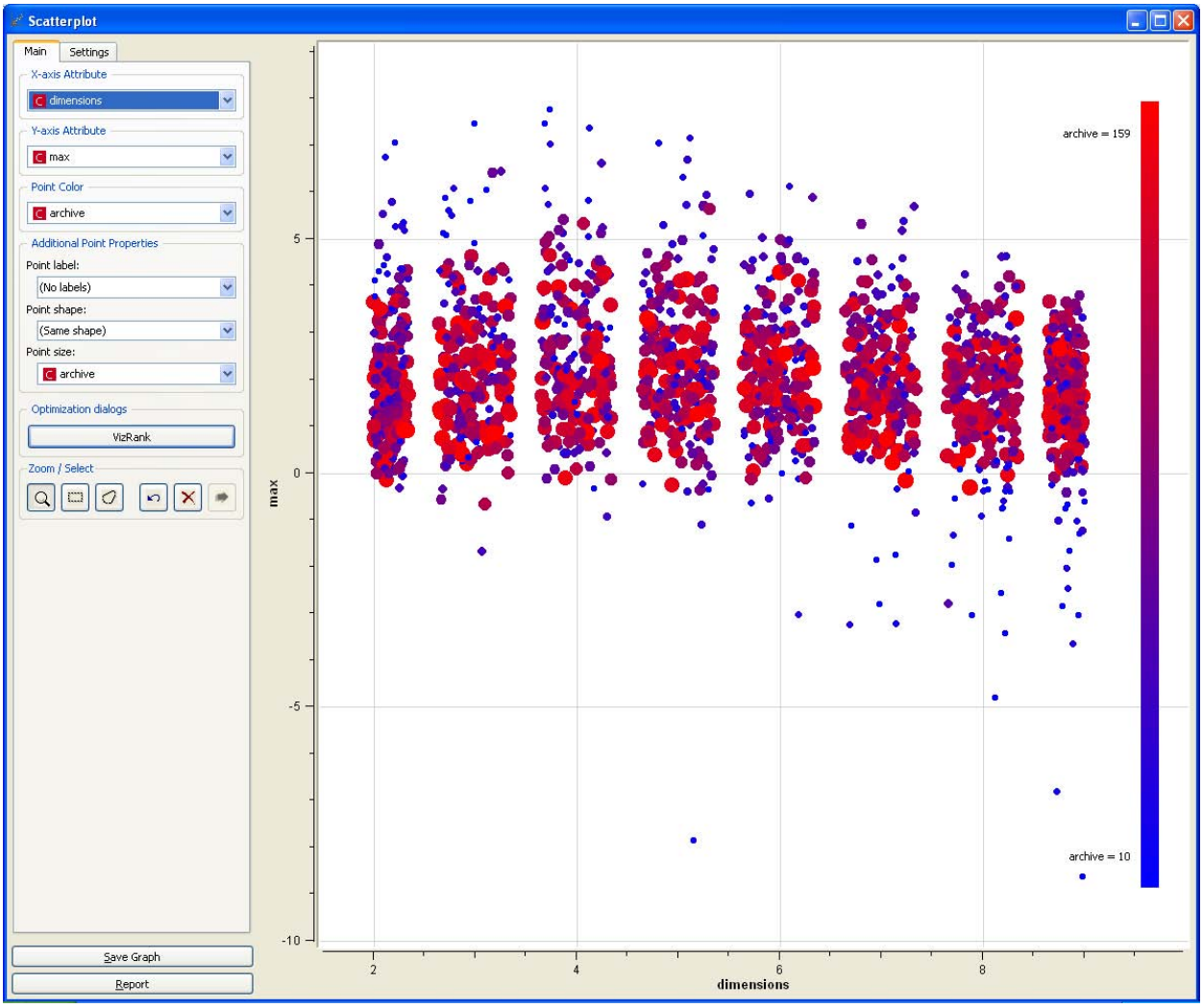


Figure 13: Dimensionality vs. Output. Student archive size determines the size and color of the circles. Small archives (small, blue circles) are better when dimensionality is small (to the left). Large archives (large, red circles) are better when dimensionality is large (to the right).

These results translate to two recommendations for selecting parameter values for MaxSolve student-test coevolution. First, the student and test mutation rates should be complimentary. The COMPARE-ON-ONE numbers game benefits in general from a high mutation rate, as mutations do not become more destructive as the individuals improve. However, the improvement in students is limited by both their mutation rate and the ability for tests to detect improvements between mutations. Second, the choice of optimal archive size depends on the problem dimensionality. This is a critical component; larger archive sizes cause more individuals and more computation, increasing running time and resource usage.

With this information, our challenge adaptation and student strategy estimation can be more effective by 1) placing equal emphasis on student and test mutation and 2) estimating the problem dimensionality and tuning the archive size. For example, given a model of the trainee's strategy, ESTATE will generate a new challenge that will defeat the trainee but still fall within the Zone of Proximal Development (ZPD), allowing the trainee to improve with practice. Using the problem tuned parameters, ESTATE invokes coevolution to improve upon the trainee's strategy until it reaches the edge of the ZPD, and then selects from the latest population of tests to present a new challenge to the trainee. When ESTATE's coevolution is more efficient due to our tuned parameter selection, ESTATE can perform this function for a wider class of skill sets and strategies as well as return results faster and more reliably.

3.3 Task 3: Enhance Adaptation Techniques

The purpose of this task is to provide off-line challenge adaptation to best meet a trainee's current training needs. ESTATE performs this task by 1) estimating the current ability of a trainee and 2) generating a challenge within the Zone of Proximal Development (ZPD). The techniques used to mitigate disengagement by estimating the trainee's current ability can be reapplied to this problem. For this reason, much of the work performed for Task 2 also applies to Task 3.

We are currently investigating techniques to estimate the ZPD. Because training rates vary across application domains, the size of the ZPD on a particular application may not be known ahead of time. Our method of continuously estimating the ICC curve during learning (see Section 3.2.1.6) may be applied toward estimating a ZPD. In this instance, the ZPD is a proportion of the total range of skill. For our coevolutionary strategy representation, the ZPD may be a measure of how much the trainee's strategy can be expected to change to defeat a particular challenge. This change can be a distance metric between strategies or a measure on the coevolutionary algorithm, such as number of generations needed to construct a winning strategy.

3.4 Task 4: Develop Trainee Model Processing.

The purpose of this task is to estimate a trainee's ability based on trainee performance on the given challenges. The techniques used to mitigate disengagement by estimating the trainee's current ability may be reapplied to this problem. However, to ground our trainee models we have investigated data gathered from students attempting the MoneyBee activity. We characterize the students' performance over time compared to our estimate of problem difficulty, showing that a student performance improves on average as they attempt more problems-they are able to

complete more difficult problems in less time. We have also begun creating visualizations of the student strategies to see how strategies adapt as the students attempt more problems.

3.4.1 Investigation of MoneyBee Data

MoneyBee is a coin algebra activity. The student is given a sum and a number of coins and has to pick out which coins add up to the sum. A session consists of paired exercises until a student solves five challenges. In each exercise, a student creates a problem for the other to solve, the other student receieves the problem and works on it using a graphical workbench. If the student solves the problem in the alloted time, both students receive points according to the problem difficulty. The MoneyBee activity is an example of a human managed tailored training activity. The students are incentivized to present the most difficult problem they believe the other can solve.

In the direction of our work on Item Response Theory, we attempted to establish an independent heuristic that could predict the difficulty of a particular MoneyBee problem. Such a heuristic may be able to inform the creation of a Zone of Proximal Development for similar challenge sets to identify challenges which are more difficult but still within the trainee's ability.

Our initial heuristic performs the following calculation to estimate difficulty. Begin with the initial amount of cents:

1. Remove the odd pennies (modulo five)
2. Search for the solution adding a single coin in a breadth first search (first quarters, then, dimes, then nickels, then pennies), until the problem has only one coin type remaining.
3. The logarithm of the number of steps in the search is the difficulty rating.

This heuristic makes the assumption that players will attempt larger valued coins first, and that players mentally search for a solution by considering all alternatives in sequence. Because breadth first search is exponential in the number of nodes explored, the logarithm of the heuristic is the estimate.

The first step is to validate this difficulty rating heuristic against the average times taken to complete the challenges. Figure 14 shows the results of plotting the log scaled heuristic against the time taken to complete the problem. As the regression line shows, there is a positive correlation between the heuristic and the time to completion.

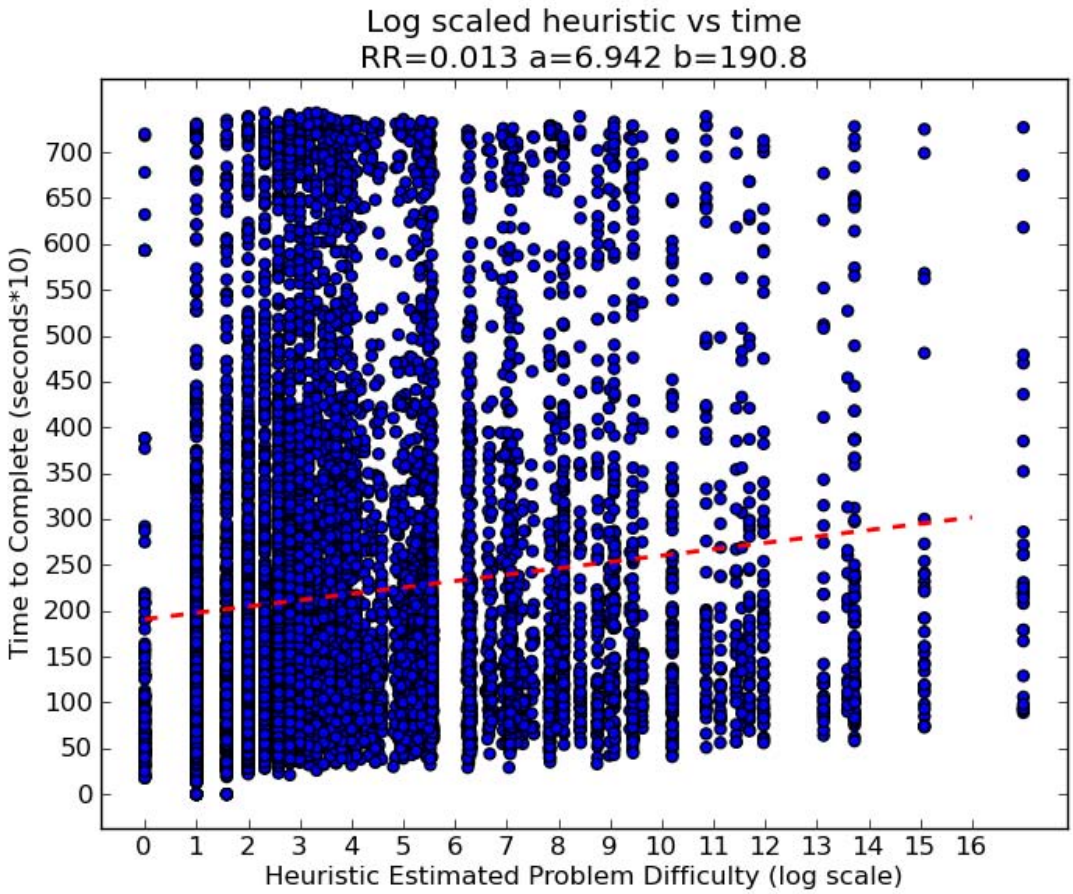


Figure 14: Log Scaled Heuristic vs. Time to Completion.

Next, we wish to discover if the players appear to be learning from repeated attempts at the challenges. Figure 15 shows a graph of the estimated problem difficulty per session. As students play more sessions they are given problems with higher estimated difficulty. Thus, as students play more sessions their partners estimate that they will be able to solve more difficult problems. Figure 16 and Figure 17 show the relation between number of sessions played and mean and median time to completion. As students play more sessions their time to complete each game decreases, indicating that they are able to solve these problems with more proficiency. Together, these analyses indicate that students are learning through challenges, solving more difficult problems in less time as they gain experience.

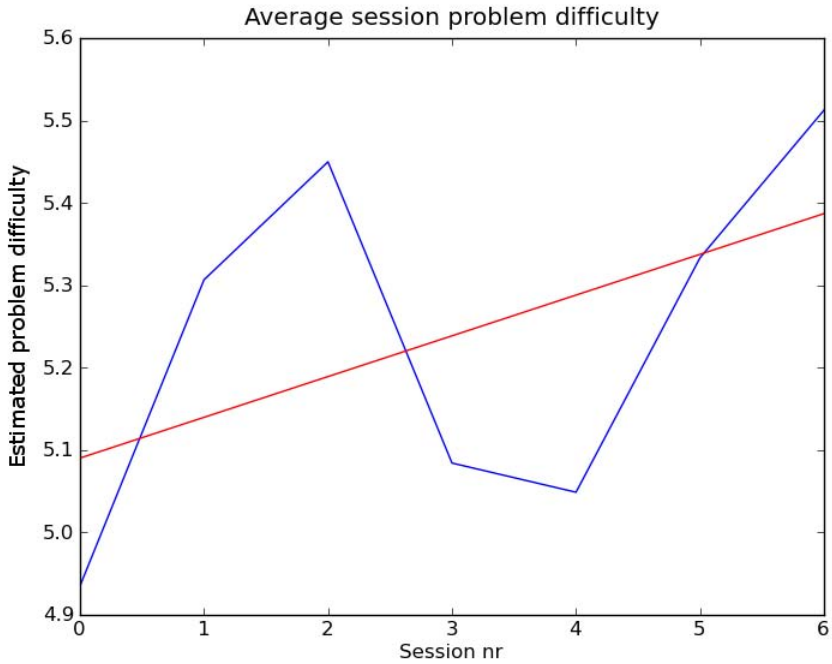


Figure 15: Estimated problem difficulty per session.

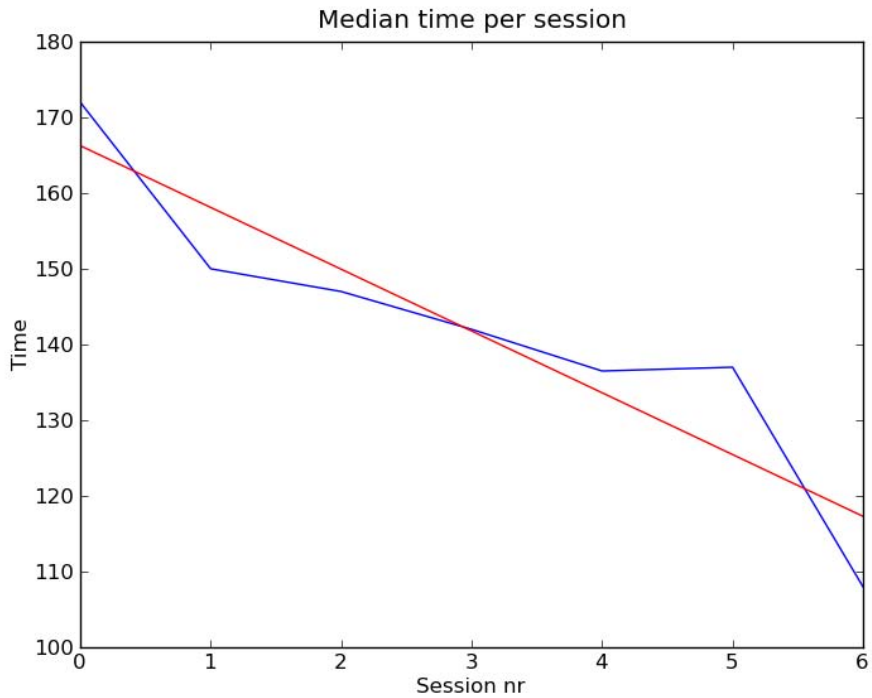


Figure 16: Median average game time per session

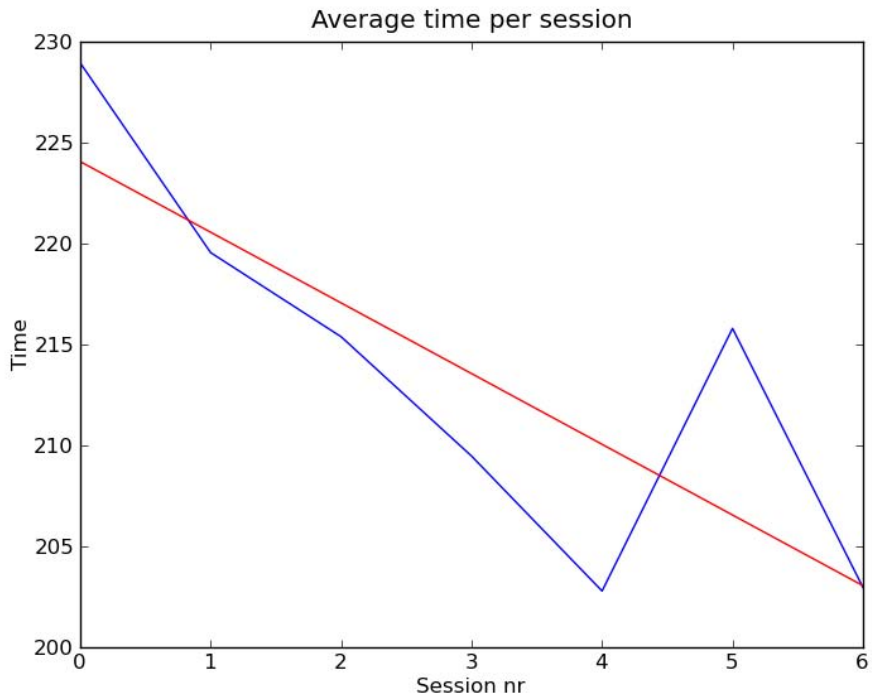


Figure 17: Mean average game time per session

3.4.2 MoneyBee Player Strategy Visualization

To improve our understanding of how trainees may employ strategies to approach difficult challenges, we created visualizations of the choices made by players of the MoneyBee game. Our visualizations are graphs of nodes that show how players move through the states of the game by making a choice at each state.

Figure 18 is a close-up view of one such player strategy graph. Each node has 6 fields. The top field is the coin state in the order of quarters, dimes, nickels, and pennies and the bottom fields are the percentages of quarters, dimes, nickels, pennies percentages selected at that state. The top node in Figure 18 is a game state with 1 quarter (represented by 1000), and was arrived at by the selection of a quarter 47% of the time, a dime 41%, a nickel 6%, and a penny 6%. The edges indicate the previous states. In Figure 18, 1000 was followed by 1100 and 1010.

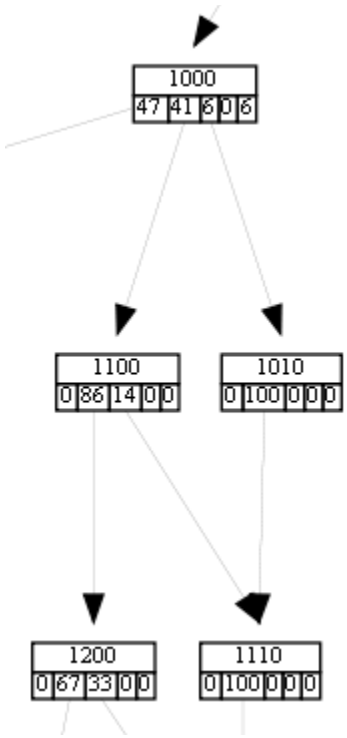


Figure 18: Detailed of player strategy graph from problem 1332

Figure 19 shows the player strategy graph of the entire problem of 1322. The two largely disconnected sub graphs indicates that two major strategies have been used on this problem, but one of them is unsuccessful, requiring the player to either backtrack or fail. The node at which these strategies diverge is a key decision point for this problem, and may represent an important concept to practice during training.

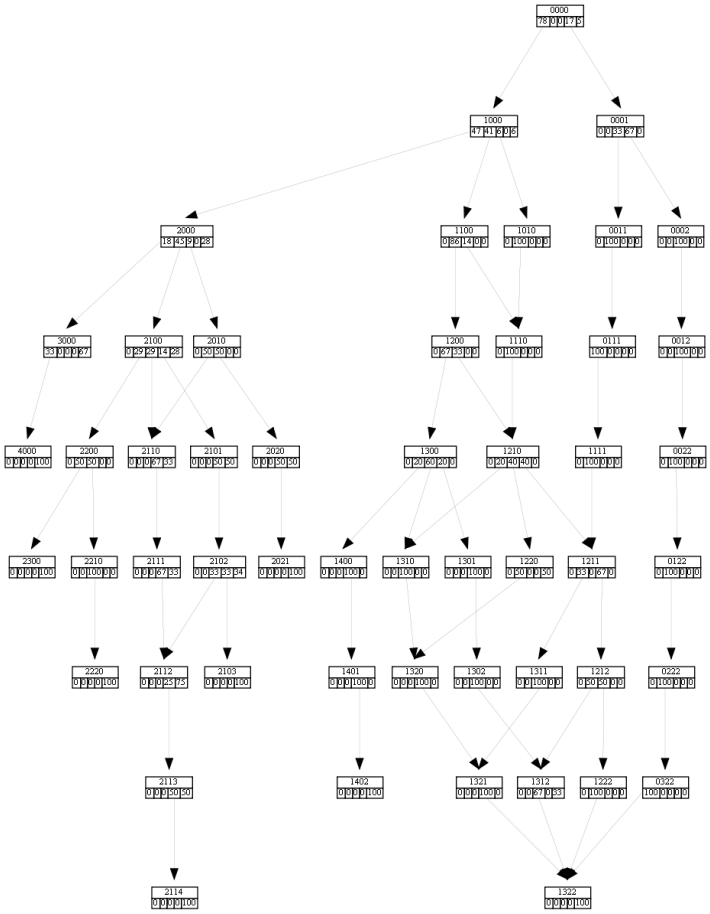


Figure 19: Entire player strategy graph of problem 1322

To understand the strategies employed, we improved upon our graph-based visualization of the strategy space to focus on perceptual methods for presenting the paths taken by players through the game state space. These visualizations are presented in Figure 20 through Figure 23 in Appendix A.

Each of the visualizations presents a single representative challenge problem in the MoneyBee dataset. In this case, the challenge problem faced is 8 coins that add up to 82 cents. The correct solution is 2 of each coin: 2 quarters, 2 dimes, 2 nickels, and 2 pennies. Each game state is represented by a node with a four digit number (QDNP), signifying the number of quarters, nickels, dimes, and pennies.

Our visualization technique uses a combination of node color, node brightness, and edge thickness to perceptually reveal elements of the strategy space. First, each node in the game state graph is color coded. Green nodes indicate valid states on the way to the correct solution state. Yellow nodes have gone over the number of coins needed, but still are below the target coin

value. Red nodes have violated both conditions, where the number of coins and the current coin value you are above the solution state. The Blue node is the goal state. Second, each node's brightness is determined by how often that node is visited during play. Nodes that are visited less are scaled darker. Bright green nodes, therefore, are visited most often. Third, an edge is placed between the transitions between game states, either by the player adding a coin (black arrowhead), removing a coin (gray arrowhead), or resetting the game (arrow to 0000 node). The width of each edge is scaled based on the frequency of how often that transition occurs.

Each visualization is constrained to players who were faced with this problem during a particular session. For example, Figure 20 shows the outcome for players who encountered this problem during their first session, or early on in their learning process. Figure 23 shows the outcome for players who encounter this problem in their fourth session, meaning they had encountered more problems before this. Based on our previous analysis, we had discovered that players do perform better in later sessions over earlier sessions.

3.4.3 Analysis of Strategy Visualizations

Several trends emerge when comparing the visualizations across sessions. First, the number of game states explored rapidly decrease, indicating that novice players are inconsistent among one another while expert players develop common strategies. Second, the number of game states visited with violations also decreases, indicating that expert players can preemptively or reactively identify violating states and recover from them gracefully. Third, there is a decrease in the amount of backtracking or resets. Finally, it is clear that explicit dominant strategies emerge early on and grow stronger in later sessions.

Now, let us look at each session individually. Figure 20 displays the results for players who encountered this challenge during their first session. While the number of states visited is large, dominant paths emerge. Specifically, there are four dominant paths that emerge. One dominant path follows the path of adding two of each coin in sequence, starting from the largest denomination to the smallest denomination. In other words, adding two quarters, then two dimes, then two nickels, and two pennies. Alternatively, the other dominant paths begin follow the same initial path before diverging. In this case, one of each coin is added in sequence, starting from the largest denomination to the smallest denomination, arriving at the 1111 game state. From here, the path diverges equally into three directions. One direction is to repeat this process, adding one of each coin in sequence, starting from largest denomination to the smallest. Alternatively, another path repeats the initial process, but adds from the smallest denomination to the largest denomination in sequence. The final path can adds one coin of increasing denominations, but

begins at the nickel, followed by dime and quarter before finishing with the penny. All three of these strategies can be seen as variants of a higher-level strategy, that of adding one of each coin in a sequence (largest to smallest denomination) followed by a repeat of this process (largest to smallest or smallest to largest denomination).

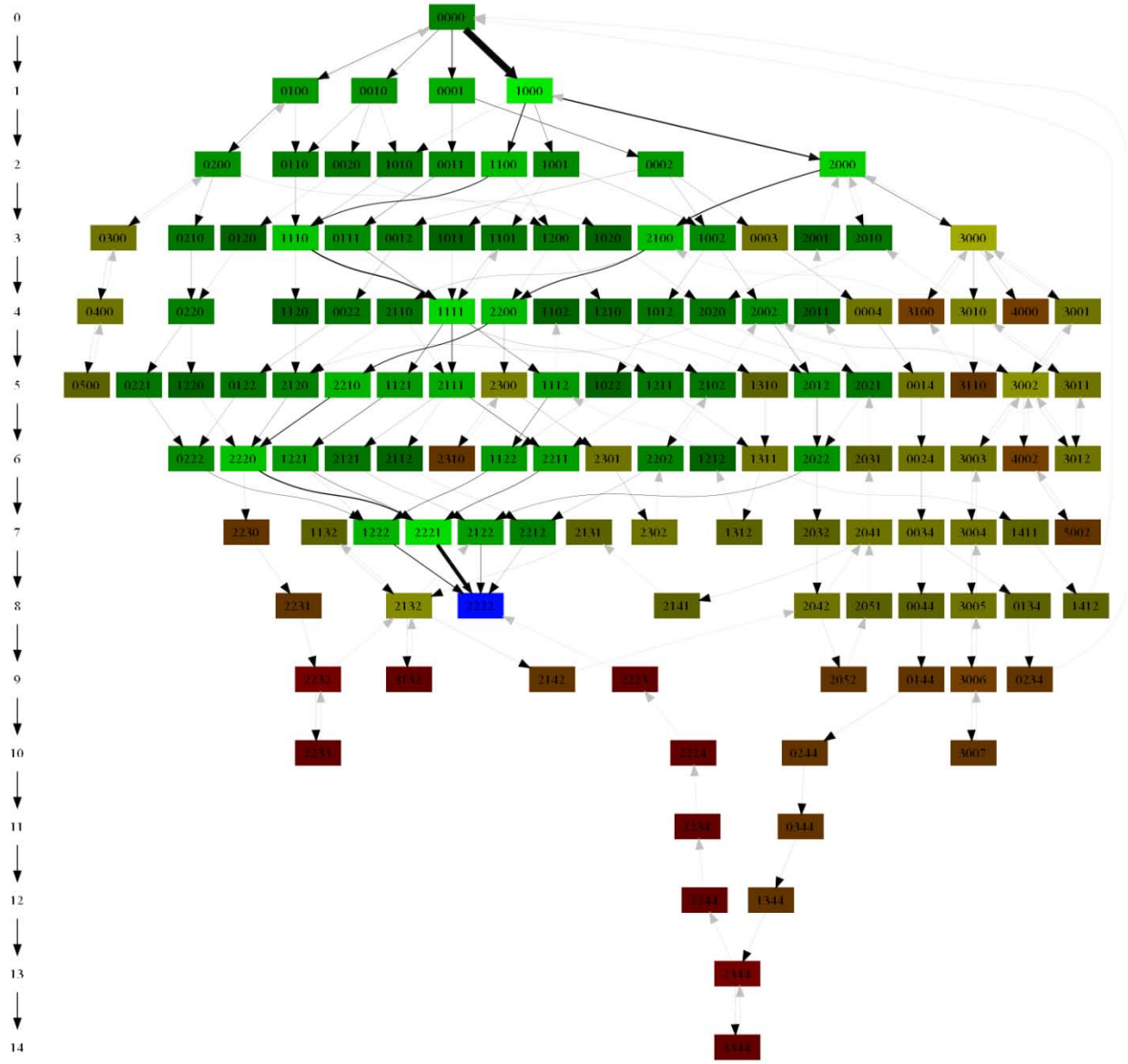


Figure 20: Strategy Visualization for Session 1 Games

In addition to the two dominant strategies, there are other major features of the data for first session games. First, players move into violating game states. Both in going over the number of coins (yellow nodes) and going over the target total amount (red nodes). Furthermore, backtracking is evident (indicated by the number of grey arrows moving up the tree), including

full resets (grey arrows from advanced game states back to the 0000 game state) when violations are identified. Finally, some traces indicate circuitous routes to the goal state, using more coins indicated than necessary and then removing them to arrive at the goal state.

There is noticeable difference in the results of traces of second session games of this challenge when compared to the first session games. As illustrated in Figure 21, adding a quarter as the first move is much more prevalent. Additionally, the two of each coin in decreasing denomination is the dominant strategy. However, closer inspection does illustrate the alternative strategy of one of each coin in decreasing denominations, however, once arriving at the 1111 game state, the strategies split equally between adding one of each coin in decreasing denominations (1111 → 2111 → 2211 → 2221 → 2222) or increasing denominations (1111 → 1112 → 1122 → 1222 → 2222). This seems to indicate that more experienced players have developed some common strategies for attempting problems. Additionally, there are fewer violations, much less backtracking, a limited number of resets, and less circuitous routes to the goal state.

Moving to third session games, as shown in Figure 22, reveals a major shift. There is a major reduction in the number of paths taken. The number of branches at a given node is often only one, indicating that players either (a) have a plan in mind when making a move, or (b) can identify the next best move at each game state. The quarter-first move still dominates and the "add two of each coin in sequence from largest denomination to smallest denomination" strategy is noticeable. The interesting feature of the third session games is that no violating states are visited nor is there any backtracking.

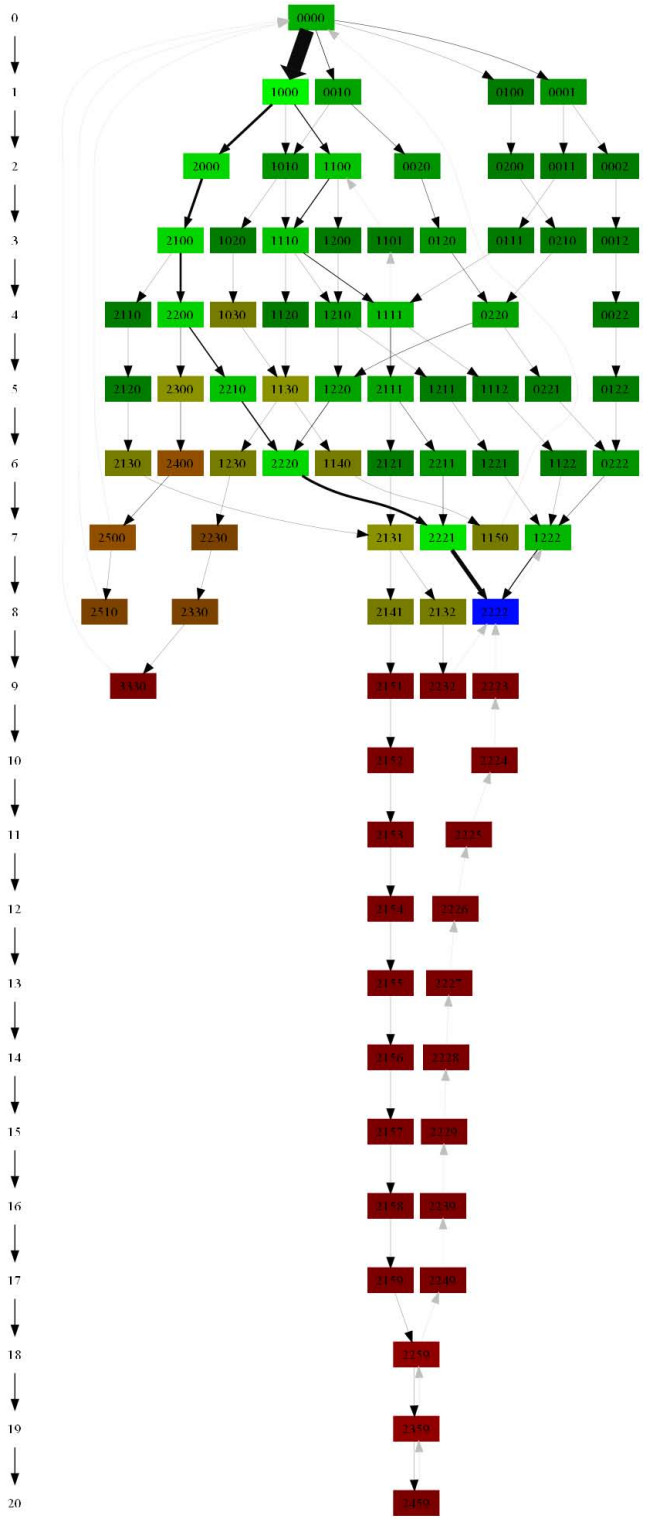


Figure 21: Strategy Visualization for Session 2 Games

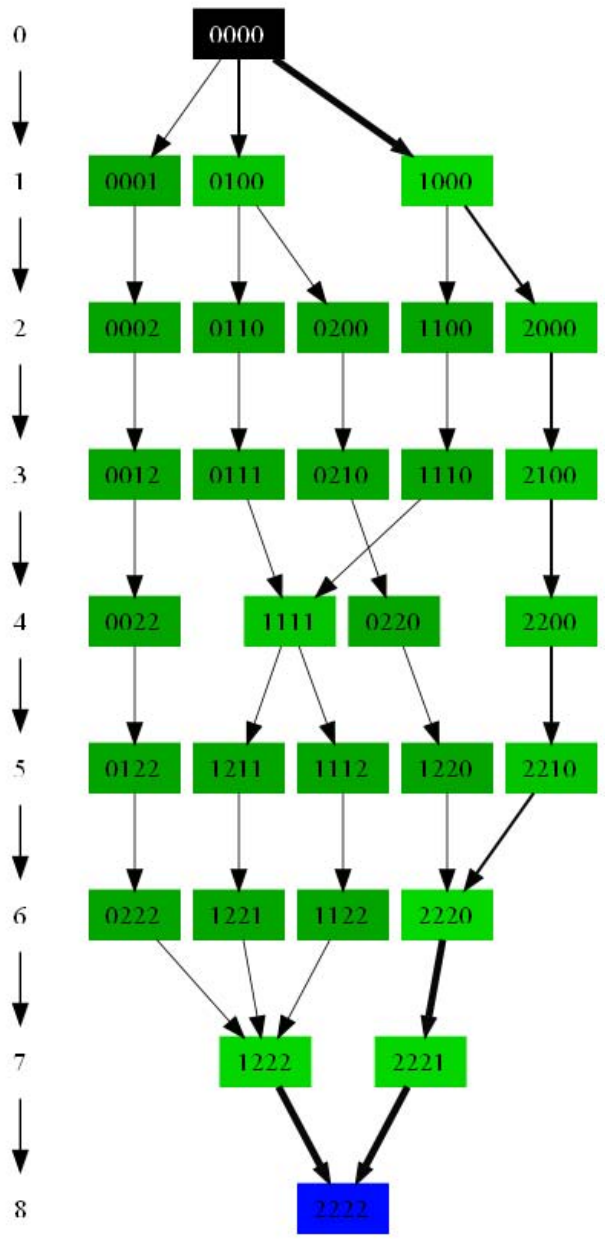


Figure 22: Strategy Visualization for Session 3 Games

The fourth session games in Figure 23 exhibit a dramatic result. Adding a quarter first (and second) is the only initial move. At this point, the "add two of each coin in sequence from largest denomination to smallest denomination" is the dominant strategy. However, we do see some deviation in some traces, backtracking when entering violating states or pressing forward and removing the violation to arrive at the game state. However, the state space of visited nodes is

dramatically small and driven in primarily one path. This indicates that experienced players converge on a dominant strategy.

This visualization method has proved instrumental to analyzing the strategies employed by players with various levels of expertise. Our next step would be to analyze other challenge problems to develop a common set of strategies by players that we can model for experimentation purposes. We can then perform experiments using our simulated trainee models to present challenges that will push players to adapt their strategies.

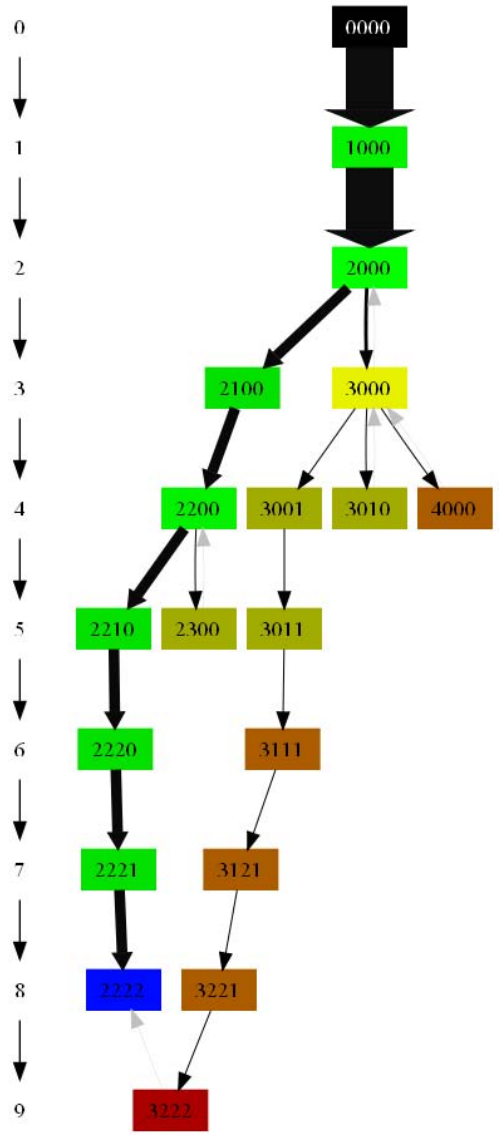


Figure 23: Strategy Visualization for Session 4 Games

3.5 Task 6: Simulation-based Training System Integration

we are now in the process of selecting and implementing challenge domains to evaluate the ESTATE approach. We considered constructing toy domains that match the abstract challenge games previously used to simulate ESTATE performance, such as a maze type game to simulate a challenge tree. These toy domains are advantageous in that they may be quickly implemented and evaluated. However, they lack depth and may not be representative of the structure of actual domains to which the ESTATE approach may be applied. Therefore, we elect to integrate ESTATE with an existing Charles River Analytics project with a well defined challenge domain and a need for adaptive training.

During the current reporting period, we have begun integration design and implementation with an ongoing Charles River Analytics effort, Pictorial Representations of Medical Procedures to Train for Effective Recall (PROMPTER). PROMPTER is funded by the U.S. Army Aeromedical Research Laboratory (USAARL) under Government Contract W81XWH-09-C-0049. PROMPTER uses an intuitive, standardized symbology to represent first-aid tasks, a pictorial mnemonic framework to visually represent first-aid procedures, and a microgame-based training method to improve comprehension and recall of the procedures. However, PROMPTER currently lacks significant adaptive training capability; the choice of challenges in the microgame is random or according to a hand-coded estimation of difficulty. Charles River Analytics will use experiments with human participants to evaluate the PROMPTER approach. Therefore, the ESTATE effort may directly benefit from this integration by implementation within the PROMPTER microgame training framework and possibly as a component tested during the human participant experiments. The PROMPTER effort may directly benefit by using the adaptive training technology in ESTATE to improve training outcomes.

3.5.1 PROMPTER Overview

Problem

Historically, the U.S. Armed forces have aggressively sought ways to reduce battlefield fatalities. Advances in evacuation techniques and personal protective equipment are two examples of this approach. However, reducing combat fatalities still demands quick and effective emergency care on the battlefield. The responsibility of providing this care does not fall exclusively on the shoulders of highly trained combat medics. All Soldiers—regardless of their medical background or experience—must be capable of providing immediate, basic first-aid to themselves (“self-aid”) or comrades (“buddy-aid”) to address a range of critical, but treatable, combat injuries (e.g., hemorrhaging in an extremity). A number of emergency medicine

technologies that address these injuries have been recently developed and deployed with the intent of reducing preventable mortality rates. These technologies include new tourniquet designs (Walters et al., 2005) and advanced hemostatic dressings (Pusateri et al., 2003). However, even with such technologies, successful treatment outcomes still require those performing first-aid to rapidly select and effectively execute an appropriate response procedure, all under considerable time pressure in a chaotic battlefield environment. To this end, all Soldiers are required to maintain proficiency for seventeen critical first-aid procedures described in the Soldier's Manual of Common Tasks, Warrior Skills, Level 1 (STP 21-1-SMCT, 2007).

While seventeen may seem a small number of tasks, training Soldiers to rapidly and effectively recall emergency medical procedures in dynamic, highly stressful, and life-threatening battlefield environments remains a challenge. This is due in part to the relative complexity of the procedures themselves, as each first-aid skill is composed of numerous, interrelated subtasks and processes. For example, the single procedure "Perform First-aid for a Bleeding and/or Severed Extremity" (081-831-1032) involves nearly 50 unique steps, divided across three potential wound dressing methods (emergency bandages, chitosan dressings, or field dressings) and two possible tourniquet devices (Combat Application Tourniquets (CAT) or improvised tourniquets). Often, individual subtasks require the Soldier to perform assessments and make rapid decisions that have downstream effects on appropriate treatment (e.g. "Elevate the injured part above the level of the heart, unless a fracture is suspected and has not been splinted"). Successful treatment outcomes require not only the correct performance of individual component tasks (e.g., inserting an intravenous catheter, applying a dressing, administering an injection), but also an awareness of the interdependencies and ordinal relationships between these component tasks as part of the overall procedure. Training Soldiers to become sufficiently aware of these many procedural subtasks and their interrelationships such that they can be immediately recalled under traumatic battlefield conditions will save lives.

Beyond the complexity of the tasks themselves, individual Soldiers vary greatly with respect to their unique skill sets, training needs, and aptitudes. For example, many Soldiers enter the Army with little to no prior experience in emergency medicine and receive less than eighteen hours of first-aid skills training before deployment (Basu, 2005). Others may have experience from serving as Emergency Medical Technicians (EMTs) or in other medical professions. After initial skill acquisition, individual Soldiers' training needs vary greatly, given their unique experiences in the field and the fact that emergency first-aid skills may be called upon very sporadically, if at all, over a particular tour of duty. To maintain sufficient proficiency over long periods of time, Soldiers must continually train and rehearse these emergency response skills and

procedures. Unfortunately, the cumbersome information delivery methods of the STP 21-1-SMCT manual (which contains nearly 100 pages of hierarchically ordered, text-based descriptions of tasks and subtasks with no imagery) do not support the efficient review of these complex emergency medical procedures. Also, this manual-based presentation format neither engages the Soldier in the active learning processes of skill rehearsal, nor is it capable of providing the Soldier with useful feedback regarding their current level of preparedness and unique training or rehearsal needs.

Given the challenges of maintaining sufficient first-aid skill competencies and the limitations of existing manual-based training materials, advanced training tools and rehearsal methods are required to enhance and maintain the Soldier's emergency medical skills. These training tools and rehearsal methods must support the depiction of complex procedures through simple, concise representations that may be easily and frequently reviewed by all Soldiers throughout their tour of duty. These representations should be designed for use with training methods that will enhance the Soldier's rapid and effective recall of complex procedures—including all critical subtasks and their interrelationships—under stressful battlefield conditions. Such training methods should not only address individual Soldiers' unique competencies and training needs, but also do so in a way that that effectively engages Soldiers in the training experience. These methods must also motivate the effective retention of procedural first-aid skills over protracted periods of time, which is crucial to reduce the number of preventable combat deaths.

Approach

Training tools and rehearsal methodologies based on visual learning (rather than verbal) of complex, interrelated task structures offer one promising approach to enhance the effectiveness emergency medical skills training and retention. For example, *pictorial mnemonic* training approaches (Estrada et al., 2007), have been demonstrated to support the recall of emergency procedures more effectively than rote memorization of text-based task descriptions. Such methods strive to create a simple visual representation of a task flow that can be remembered by the trainee as a single "chunk" of information. During task execution, this single visual image is recalled and its individual components are "unpacked" to identify critical subtasks, their serial relationships, and dependencies for performing the complex task.

One approach to representing a complex first-aid procedure within a pictorial mnemonic would be to develop a single storyboard depiction of individual subtasks being performed in series, much like the safety cards used by airlines, or procedural first-aid posters found in public buildings. These storyboards typically use a sequence of pictorially realistic illustrations of

component behaviors to describe multi-step procedures to users without the need for literacy. However, while such illustrations are appropriate to support rapid procedural *recognition*, they are poorly suited for training rapid procedural *recall*. Their relative complexity makes them difficult to memorize and recall as a single, coherent visual image. In contrast, an effective pictorial mnemonic device must represent a complex procedure through a visual structure that can be recalled as a single image, which can then be unpackaged into its constituent task components. To accomplish this, these mnemonics should leverage a simple, but intuitive symbology to represent critical subtask activities, decision points, and alternative process flows. This symbology must: (1) be appropriate to the emergency medicine domain while remaining highly intuitive to the target audience (e.g., Soldiers with potentially no medical background); and (2) support the effective combination of atomic task symbols into "roadmaps" of complex procedures that can be accurately recalled by the trainee as individual, sufficiently distinguishable visual objects.

However, for improved treatment outcomes, an intuitive visual symbology and pictorial mnemonics for representing emergency medical procedures must also be paired with advanced training methods, both to teach Soldiers how to use the symbology and mnemonics initially (to learn), and over time (to retain). Simply trading static, textual depictions of process flows (e.g. the SMCT manual) with static, visual depictions of process flows (e.g., flash cards) will not support the development of the rich knowledge structures necessary for procedural recall. Similarly, while providing visual training aids may make review of complex training materials more efficient, it will not intrinsically increase the trainee's motivation to learn first-aid, nor their engagement in the training process.

In contrast, the integration of intuitive, visual training materials with engaging microgame-based delivery methods represents a promising approach for enhancing both the efficiency and the effectiveness of procedural training. Microgames are lightweight, short duration (5-20 minute) computer-delivered games that can support learning over a broad range of platforms (e.g., desktop, laptop, PDA, or cell phone devices). These approaches are low-cost, can be updated quickly and inexpensively to incorporate new training material, and may be easily and cheaply distributed using ubiquitous web-based delivery methods. They are purposefully developed to engage the user, which improves learning transfer (Prensky, 2001) and encourages greater use of the games over time. The brief, visual nature of traditional microgames makes them well-suited to repetitive cognitive skills training, particularly for tasks related to pattern matching, memorization, and visual recall. Microgames also lend themselves to integration with intelligent, adaptive methods to continually assess training performance against pre-determined

competency goals and adaptively manipulate the type and complexity of individual microgame tasks to enhance the Soldier's skill acquisition and retention over time.

PROMPTER has previously demonstrated that combining visual task symbologies and microgames is not only feasible, but also that it represents an innovative approach to enhancing the training of medical procedures. The current PROMPTER effort is to develop and evaluate task symbologies and adaptive microgames that use pictorial representations of medical procedures to train for effective recall. The pictorial mnemonics and engaging microgame-based rehearsal methods developed and tested under PROMPTER will allow individual Soldiers to more efficiently develop and maintain the ability to rapidly recall emergency first-aid skills Four major components comprise our approach. First, we are designing an ***intuitive, standardized symbology*** for the individual first-aid task and subtasks that comprise the complex emergency first-aid skills of the Soldier's Manual of Common Tasks, Warrior Skills, Level 1 (STP 21-1-SMCT). This symbology will be designed from a human-centered perspective to be highly usable by its intended audience (ranging from new Soldier recruits with no medical background to trained combat medics), in terms of interpretability, learnability, discriminability, and simplicity. Second, we will incorporate sets of these first-aid symbols within a ***pictorial mnemonic framework*** to visually represent each of the seventeen procedures in STP-21-SMCT. This framework will support the creation of individual pictorial mnemonic devices that effectively convey the related actions of each particular procedure through a single, cohesive and highly memorable visual image. Third, we will design and demonstrate *adaptive,* ***microgame-based training methods*** that leverage these pictorial mnemonic training materials. These microgames will present tasks and challenges relevant to procedural skill acquisition and retention, using engaging game play mechanisms that are continually tailored to individual Soldiers' evolving training needs. The microgame platform and adaptive content-generation process will be both generic and extensible to support pictorial mnemonic-based procedural training across a broad variety of military and civilian application domains (e.g., aviation, process control, natural disaster management). Fourth, we will conduct formal ***evaluations*** to assess the PROMPTER training materials and methods. We plan a set of evaluations to specifically target the usability of the PROMPTER task symbology, pictorial mnemonics, and adaptive, game-based training methods, as well as their effectiveness in supporting Soldiers' learning and maintenance of first-aid skills, in comparison to traditional, text-based training materials.

Implementation

Figure 24, Figure 25, and Figure 26 show examples from the standardized symbology and pictorial mnemonic framework. These symbols and pictorial mnemonics make up the basic elements of the PROMPTER microgames. Figure 27 shows three such microgames that may be constructed with these elements. In the first (a), the trainee must choose the symbol that matches the meaning of the text. In the second (b), the trainee must choose a symbol that does not belong or is out of place in the procedure. In the third (c), the trainee must create a procedure using the individual mnemonics. During a microgame session, the trainees are presented with these individual challenges in quick succession, each lasting no more than seconds.

(a) **(b)**

Figure 24. (a) Illustration of a casualty with an abdominal wound being laid on their back with their knees bent; (b) PROMPTER icon capturing this body position through a simple, intuitive line drawing.

casualty dressing monitor/observe wound

Figure 25. Examples from core set of symbol primitives for commonly occurring objects and actions

monitor for shock | abdominal wound | place casualty's bandage on wound, white side down | pick up organs using clean material

Figure 26. Examples of compound symbols that combine core symbols of the PROMPTER visual alphabet to express more complex task concepts

(a) (b) (c)

Figure 27. Examples of rapid prototypes developed to explore and demonstrate promising mechanics for microgame-based learning of procedural knowledge structures, including games that test and develop: (a) knowledge of the meaning of individual symbols; (b) ability to visually recall overarching task mnemonics; and (c) ability to rapidly reconstruct complex task processes

3.5.2 ESTATE and PROMPTER Integration

ESTATE may use the PROMPTER microgame training platform, existing software, and experiments as a test case for the adaptive training approach. A trainee plays a session of a PROMPTER microgame, and ESTATE creates a skill model of the trainee. The skill model and new challenges are evolved using student-test coevolution until the challenges and evolved skills have reached a significant distance from the initial state (i.e., they have reached the zone of proximal development (ZPD) for the trainee). The set of evolved challenges are packaged into a microgame session for the next time the trainee logs on and attempts the microgame. At this time, the skill model of the trainee is updated and ESTATE again adapts the set of challenges for the next session. Due to ESTATE's avoidance of coevolutionary pathologies, the adaptation drives the trainee towards continuous improvement without cycling, evolutionary forgetting, overspecialization, or disengagement.

Currently, the PROMPTER implementation consists of a server backend and a javascript game client capable of running on multiple devices, including PC web browsers and smartphone platforms such as the Android and iPhone. The game clients receive game content, user profiles, and media from the server via HTTP. The client sends the actions performed by the human user to the server, where they are stored in a database. The user's performance can then be evaluated by a supervisor at a later date. Figure 28 shows a simplified diagram of the PROMPTER

architecture. The server also includes a web interface that can be used by supervisors to access user profiles and performance data. For simplicity this interface is not shown in the figure.

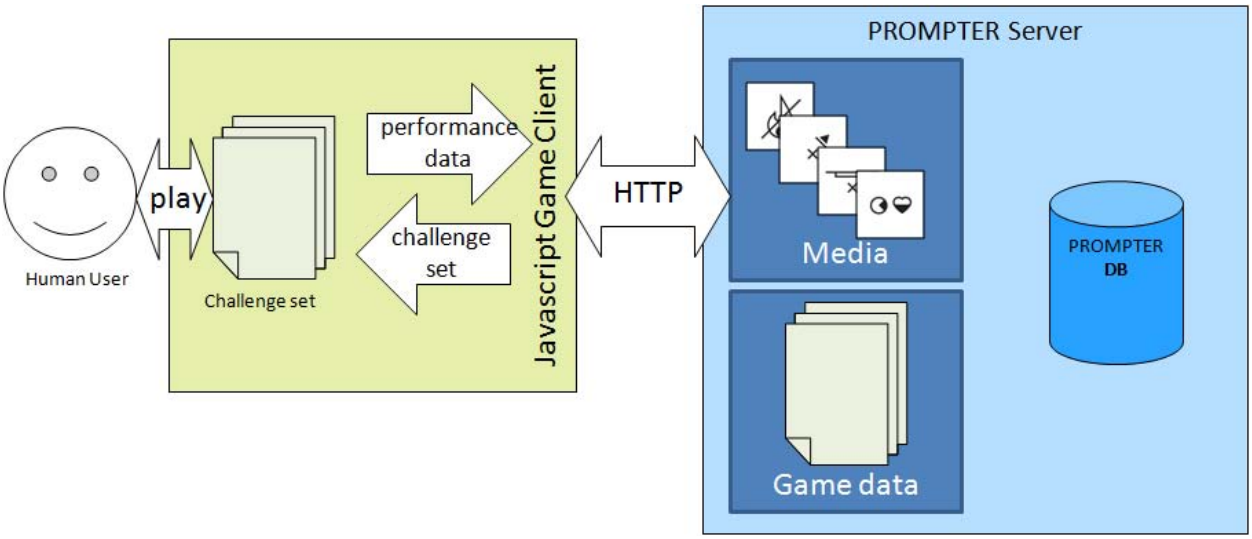


Figure 28. Current PROMPTER Architecture

The client/server model of PROMPTER lends itself towards straightforward integration and modification. To allow ESTATE's simulated players to play PROMPTER's games, modification to PROMPTER code is minimal. Performance metrics, for example, can be accessed programmatically using the server's existing interfaces, which provides trainee performance data in XML format.

To incorporate the ESTATE adaptive training techniques, the PROMPTER game clients will be replaced with a "thin" interface that communicates with the PROMPTER server using the existing messaging architecture. This interface will allow both the simulated trainee as well as the coevolution trainee models to play simulated PROMPTER games.

The PROMPTER server can be re-used with only slight modifications. Currently, PROMPTER clients have no control over which question or challenge is posed when a game is played. The ESTATE adaptive training technique requires complete control over the challenges presented as well as their ordering. This feature may be implemented by expanding the communication protocol between the server and client or by allowing the ESTATE client direct access to the server database (i.e., allowing it to seed the games with the desired challenges). Figure 29 shows the designed integration architecture for ESTATE and PROMPTER pure simulation experiments. The thin interface will lack any visible UI since the players are automated; instead, it serves to connect both the simulated human user and the evolved user

models to the PROMPTER server's games and to collect performance data. Figure 30 shows the designed integration architecture for a playable ESTATE and PROMPTER prototype.

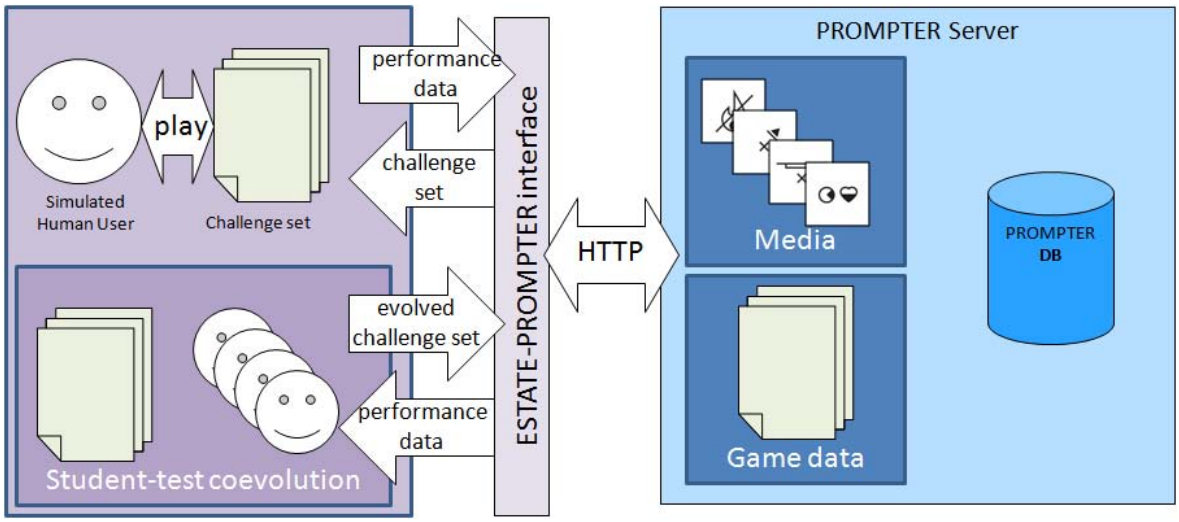


Figure 29. ESTATE-PROMPTER simulation framework integration design

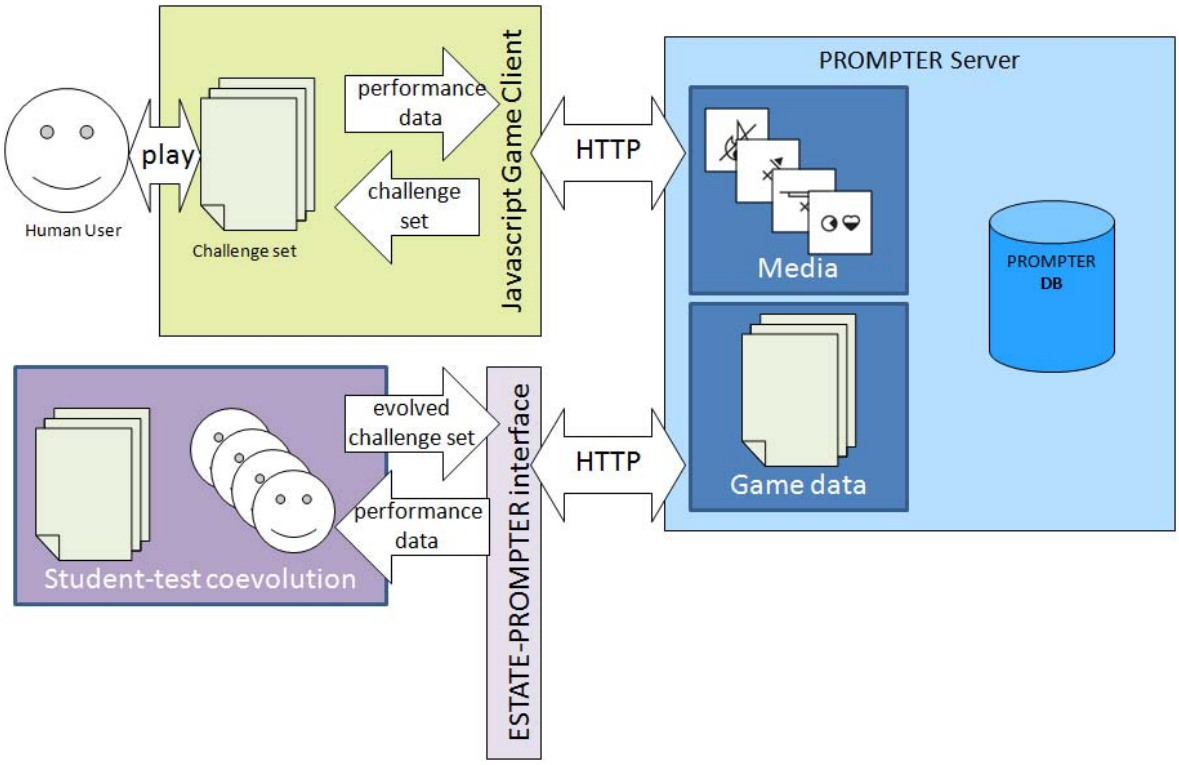


Figure 30. ESTATE-PROMPTER playable integration design

3.6 Task 8: Transition

At the two-day ONR 341 Program Review for Harold Hawkins on 4-5 October 2010, we met Dr. Kendy Vierling, a senior analyst at the Human Performance, Training, and Education MAGTF Training Simulations Division of the US Marine Corps. We are corresponding with Dr. Vierling, who has been helpful in finding opportunities for ESTATE in the USMC Training Systems Division. We are currently pursuing possibilities for adaptive cultural training in virtual environments.

4. References

Baker, F. B. (2001). Basics of Item Response Theory. ERIC Clearinghouse on Assessment and Evaluation.

Basu, S. (2005). First Aid Skills Revised For Soldiers. U.S.Medicine [On-line]. Available: http://www.usmedicine.com/article.cfm?articleID=1074&issueID=74

Bouton, C. (1901). Nim, A Game With a Complete Mathematical Theory. The Annals of Mathematics, 3(1), 35-39.

Bucci, A. (2010).

De Jong, E. (2005). The MaxSolve Algorithm for Coevolution. Washington, DC.

De Jong, E. & Bucci, A. (2006). DECA: Dimension Extracting Coevolutionary Algorithm.

De Jong, K. A. (2004). The Incremental Pareto-Coevolution Archive. In Proceedings of The Genetic and Evolutionary Computation Conference, (pp. 525-536).

Estrada, A., Keeley, J. A., Leduc, P. A., Bass, J. M., Rouse, T. N., Ramiccio, J. G. et al. (2007). A Novel Approach in Facilitating Aviation Emergency Procedure Learning and Recall Through an Intuitive Pictorial System. (Rep. No. 2007-07).

Ficici, S. G. (2004). Solution Concepts in Coevolutionary Algorithms. Brandeis University.

Prensky, M. (2001). True Believers: Digital Game-Based Learning in The Military. In Digital Game-Based Learning. Mcgraw-Hill.

Pusateri, A. E., Modrow, H. E., Harris, R. A., Holcomb, J. B., Hess, J. R., Mosebar, R. H. et al. (2003). Advanced Hemostatic Dressing Development Program: Animal Model Selection Criteria and Results of a Study of Nine Hemostatic Dressings in a Model of Severe Large Venous Hemorrhage and Hepatic Injury in Swine. The Journal of Trauma: Injury, Infection, and Critical Care, 55(3), 518-526.

Rosin, C. D. (1997). Coevolutionary Search Among Adversaries. University of California, San Diego, CA.

Walters, T. J., Wenke, J. C., Greydanus, D. J., Kauvar, D. S., & Baer, D. G. (2005). Laboratory evaluation of battlefield tourniquets in human volunteers. Storming Media.